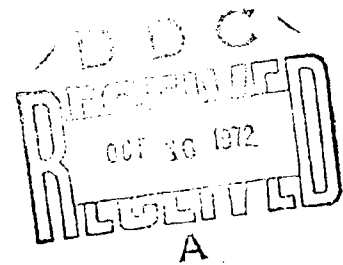AD 749686

A Fortran Computer Program
to Calculate the Range of a Pulse Radar

L. V. BLAKE

Radar Geophysics Branch
Radar Division

August 28, 1972

NAVAL RESEARCH LABORATORY
Washington, D.C.

Security Classification

DOCUMENT CONTROL DATA - R & D

(Security classification of title, body of abstract and indexing annotation must be entered when the overall report is classified)

1. ORIGINATING ACTIVITY (Corporate author)	2a. REPORT SECURITY CLASSIFICATION
Naval Research Laboratory	Unclassified
Washington, D.C. 20390	2b. GROUP

3. REPORT TITLE

A FORTRAN COMPUTER PROGRAM TO CALCULATE THE RANGE OF A PULSE RADAR

4. DESCRIPTIVE NOTES (Type of report and inclusive dates)

A final report on one phase of the problem; work is continuing on other phases.

5. AUTHOR(S) (First name, middle initial, last name)

L. V. Blake

6. REPORT DATE	7a. TOTAL NO. OF PAGES	7b. NO. OF REFS
August 28, 1972	58	16

8a. CONTRACT OR GRANT NO	9a. ORIGINATOR'S REPORT NUMBER(S)
NRL Problems R02-64 and R02-55A	
b. PROJECT NO	NRL Report 7448
Projects PO-0-0038 and SF 11-141-005-15483	
c.	9b. OTHER REPORT NO(S) (Any other numbers that may be assigned this report)
d.	

10. DISTRIBUTION STATEMENT

Approved for public release; distribution unlimited.

11. SUPPLEMENTARY NOTES	12. SPONSORING MILITARY ACTIVITY
	Department of the Navy
	(Naval Ship Systems Command)
	Washington, D.C. 20360

13. ABSTRACT

A Fortran computer program to calculate radar maximum range, written for the NRL CDC-3800 computer but adaptable to any computer with a Fortran compiler, is described. The computation follows previously established principles, with the pattern-propagation factors set equal to one, so that the range calculated is for free space in the sense that earth's surface effects are not taken into account. However, the effects of a standard atmosphere are included in the calculation. Reflection-interference effects can be separately described by utilizing the calculated free-space range as an input to computer plotting programs.

The program calculates the range for any specified probability of detection, false-alarm probability, and Swerling fluctuation case by utilizing a slightly modified subroutine written by Fehlner and coworkers of the Johns Hopkins University Applied Physics Laboratory. Postdetection (noncoherent) integration is assumed. The system noise temperature is computed including effects of galactic, cosmic blackbody, solar, and tropospheric noise, and the tropospheric molecular absorption for oxygen and water vapor is calculated for a standard atmosphere. The effect of refraction on the ray path is included in the absorption calculation by ray tracing, assuming a negative-exponential refractivity-height profile. The range of validity of the noise temperature and absorption calculations is approximately 100 MHz to 100 GHz. The computation requires a few seconds with the CDC-3800 computer.

A

DD FORM 1473 (PAGE 1)

Security Classification

14 KEY WORDS	LINK A		LINK B		LINK C	
	ROLE	WT	ROLE	WT	ROLE	WT
Radar detection						
Radar maximum range						

CONTENTS

ABSTRACT

A Fortran computer program to calculate radar maximum range, written for the NRL CDC-3800 computer but adaptable to any computer with a Fortran compiler, is described. The computation follows previously established principles, with the pattern-propagation factors set equal to one, so that the range calculated is for free space in the sense that earth's surface effects are not taken into account. However, the effects of a standard atmosphere are included in the calculation. Reflection-interference effects can be separately described by utilizing the calculated free-space range as an input to computer plotting programs.

The program calculates the range for any specified probability of detection, false-alarm probability, and Swerling fluctuation case by utilizing a slightly modified subroutine written by Fehlner and coworkers of the Johns Hopkins University Applied Physics Laboratory. Postdetection (noncoherent) integration is assumed. The system noise temperature is computed including effects of galactic, cosmic blackbody, solar, and tropospheric noise, and the tropospheric molecular absorption for oxygen and water vapor is calculated for a standard atmosphere. The effect of refraction on the ray path is included in the absorption calculation by ray tracing, assuming a negative-exponential refractivity-height profile. The range of validity of the noise temperature and absorption calculations is approximately 100 MHz to 100 GHz. The computation requires a few seconds with the CDC-3800 computer.

AUTHORIZATION

PROBLEM STATUS

A FORTRAN COMPUTER PROGRAM TO CALCULATE THE RANGE OF A PULSE RADAR

INTRODUCTION

The program to be described performs a calculation of the free-space range of a monostatic pulse radar, following the principles presented in NRL Reports 6930 and 7010 (1). The range calculated is "free space" in the sense that the effects of the earth's surface, such as reflection-interference and below-the-horizon shadow, are not taken into account. However, the absorption and noise of a standard atmosphere, galactic noise, and solar noise are taken into account. For detailed definitions of some of the quantities involved in range calculation and for a discussion of the theory, the reader is referred to those reports. In NRL Report 6930, a Range-Calculation Work Sheet was presented to systematize the calculation and thus to simplify handling the rather large number of quantities and computational steps involved. The use of a computer program represents a still further step toward simplification of the calculation, and also minimizes the possibility of error. The computation requires punching one card with the input data (radar parameters and related quantities). To guard against error in punching this card, the program prints out all of the input data as well as the calculated results. The program has been given the Fortran name RGCALC.

Specifically, this program is a Fortran formulation of Eq. (12) of NRL Report 6930 (Eq. (3) of this report), with the pattern-propagation factors omitted. The signal at the target is therefore assumed to be due solely to direct-path propagation, and if the specified transmitting and receiving antenna gains are those of the beam maxima, the target is assumed to be in the beam maxima. As discussed in NRL Report 6930, the equation is based on the assumption that the detection range is limited by the normal system noise — i.e., that there is no interference from manmade signals or noise, and no clutter signals caused by echoes from extraneous targets, such as rough sea or terrain, rain, or any profusion of other targets in the vicinity of the target whose detection is being considered. The range thus calculated may be called the "basic" detection-range capability of the radar.

The Range-Calculation Work Sheet of NRL Report 6930 requires some auxiliary calculations and the use of some sets of curves to determine the visibility factor (minimum-detectable signal-to-noise ratio); the antenna, transmission-line, receiver, and system noise temperatures; and the atmospheric absorption loss. Calculations equivalent to using these curves and auxiliary calculations are performed within Program RGCALC. The only auxiliary calculation required is that of number of pulses on target, for a scanning radar. It was not considered feasible to do this calculation in the computer program because the number of pulses is sometimes determined by a signal processor rather than by the scanning. However, the calculation is not difficult, in the scanning radar case, as will be discussed later in this report. Postdetection (noncoherent) integration of the pulses is assumed.*

---

*See Ref. 1 (NRL Report 6930), p. 18.

The computer program can also be adapted to calculating the range of a CW radar and of a bistatic radar by suitable redefinition of some of the input parameters. For CW radar, the pulse length τ can be interpreted as the effective sampling time of the detection processor in microseconds; or if such a processor is not used, the parameter τ can be interpreted as the reciprocal of the receiver predetection bandwidth in megahertz, or as the length of time a scanning beam remains on the target, whichever time is the shorter. The bandwidth correction factor can be used as a correction for non-optimum processing or filtering of the signal. The "number of pulses integrated" should be set equal to 1. The transmitter power P_t is defined for this calculation as the average transmitted power in kilowatts.

If the radar is bistatic, the only reinterpretation required is in the meaning of the calculated range number. Instead of representing the monostatic range, it represents the square root of the product of the transmitter-to-target range R_t and the target-to-receiver range R_r. In other words, the calculated number is the geometric mean of R_t and R_r. The transmit and receive antenna gains are already specified separately in the program because they are actually different even for some monostatic radars. The target cross section σ must of course be the bistatic value.

The frequency range over which the program may be used is from about 100 MHz to 100 GHz, but it could be extended downward to about 30 MHz and upward to perhaps 150 GHz without incurring gross errors. (Below 30 MHz the occurrence of ionospheric effects and above 150 GHz the multiplicity of water-vapor absorption resonances invalidate the equations used in the program.)

The program is written with the option of calculating range for either specified detection and false-alarm probabilities, or for a specified signal-to-noise power ratio (expressed in decibels). The latter option is useful in calculating the maximum range of a tracking radar (as distinct from a search or acquisition radar) when the minimum signal-to-noise ratio for successful tracking is known.

Except for the fact that the external noise from celestial and terrestrial sources and the absorption that occurs in the earth's atmosphere are taken into account, the free space range of the radar is calculated. Actually it would be more accurate to call this range a quasi-free-space range because of the inclusion of celestial and terrestrial noise and absorption effects. Because the absorption by the troposphere is dependent on the elevation angle of the ray path, the target elevation angle is one of the input quantities for the range calculation. The radar is assumed to be located at or near the earth's surface — within say a thousand feet of sea level. Range calculations applicable to extraterrestrial locations (e.g., satellite or space-ship radars) can be made by setting the elevation angle to a high value, e.g., 90 degrees. At this elevation angle, the absorption is usually negligible for frequencies appreciably below the 22-GHz water-vapor resonance line. Also, the computed absorption, in decibels, is printed out, so that the computed range can be revised to correct for it if a true free-space range is desired. The ray path for the absorption calculation is computed by a ray-tracing algorithm, assuming an exponential decrease of the refractivity with height (CRPL Exponential Atmosphere), with surface refractivity of 313 N units (2).

The principal non-free-space factor not taken into account is the effect of the reflection and absorption by the earth's surface. These effects may modify the free-space range greatly under some conditions, but they cannot be readily taken into account by a single

range calculation. To depict the non-free-space performance of a radar, what is needed is a graphical representation of the detection-range contours in a vertical plane, or, for a target at constant altitude, a plot of the variation of signal strength with range, relative to the signal required for detection.

Programs to produce such plots have been written, and have been described in a separate report (3). One of the inputs required for these programs is the free-space range of the radar. Hence, the present program supplements the plotting programs. They could be combined into a single package. However, this has not been done because ordinarily it is not objectionable to calculate the free-space performance and to plot the non-free-space detection curves as two successive operations.

For radars whose antenna pattern is a narrow elevatable beam, no significant reflection-interference effects occur when the beam is elevated by one beamwidth or more, and the quasi-free-space range calculation then applies directly.

The program to be described has been used and extensively tested over a period of time. Results agree with manual calculations using the range-calculation worksheet of NRL Report 6930 (1). The program was used to compute the ranges of actual Navy radars for a forthcoming NRL Radar Division report.* The execution time for a single radar range calculation, for all five Swerling fluctuation cases, is approximately 2 sec on the NRL CDC-3800 computer,† not including compilation time which is about 100 sec. (Compilation can of course be avoided by having an "object deck" or machine-language deck punched, and using it instead of the Fortran or "source deck.")

COMPARISON WITH OTHER PROGRAMS

Some years ago, a computer program was developed for the calculation of maximum radar rays by a contractor‡ for the Scientific and Technical Intelligence Center of the Office of Naval Intelligence (ONI-STIC-50). The work was completed about 1966. This program was based on NRL Report 5868, an earlier edition of NRL Report 6930 (1). It utilized some curves published in that report by reading a finite number of data points into the computer and interpolating between them; this was done for the "visibility factor" (minimum-detectable signal-to-noise ratio), the antenna noise temperature, and the atmospheric absorption loss. Because of this the program was limited to calculating the range for 0.5 probability of detection for a nonfluctuating target, and to the frequency range 100 MHz to 10 GHz.

Another computer program has been described by Boothe (4). This program computes the probability of detection as a function of the range, rather than computing

---

*This will be the fifth edition of NRL Report 5637, 4th ed., June 21, 1961, "Navy Radar Systems Survey," R.D. Tompkins.

†The execution time depends partly on the number of pulses integrated and on whether or not other immediately preceding calculations have been made for the same number of pulses and the same false-alarm probability.

‡The contractor was Control Data Corporation, and the programming was done by Mrs. Irma Wachtel. Consulting assistance on the radar-equation aspects of the work was furnished by the author of this report. The project was initiated and monitored by LCDR William Barron of ONI-STIC. The program is described in an internal ONI document STIC-CW-05-1-66 titled "Computer Program R-50, Radar Maximum Range Calculation," Feb. 23, 1967.

L. V. BLAKE

the range for a specified probability. However, the maximum range for specified probability is found by computing probability for decreasing range values until the desired probability is reached. The program also makes use of atmospheric absorption curves from NRL Report 5868, entered as data into the computer; consequently it is limited to the frequency range 100 MHz to 10 GHz, plus perhaps a few "spot" frequencies up to 100 GHz. The report does not state the method used for evaluating antenna noise temperature, and a program listing is not given. The effect of target aspect variation on cross section is taken into account deterministically, rather than statistically using Swerling's theory. The signal-to-noise ratio and the resulting probability of detection are calculated at ranges that decrease in steps corresponding to observation of a target approaching the radar. The target is assumed to be changing aspect according to some known prescription, during this approach, and the corresponding cross-section variation is calculated. (A missile target is assumed in Boothe's analysis.) As the target approaches, when the probability reaches the specified value, the range is printed out or otherwise recorded. Either single-scan or cumulative probability can be specified.

D. M. White has described a comprehensive program (5) to analyze radar performance in a dynamic situation, computing signal-to-noise ratio and detection probability as a function of time and target position, taking into account the effects of multipath interference, clutter echoes from the sea or rain, and target motion. In short, the program simulates in as much detail as is practical the complete radar-target engagement, for a single target. This program utilizes a subroutine written by L. F. Fehlner* of the Johns Hopkins Applied Physics Laboratory, and described by Fehlner in a previous report (6), to calculate the probability of detection for a specified signal-to-noise ratio, false-alarm number, number of pulses integrated, and target fluctuation characteristic. Any one of five fluctuation cases can be specified: the nonfluctuating case and the four Swerling fluctuation cases (1). White mentions other programs that have been written by Kirkwood (7) and by Nolen (8).

Killinger (9) has developed a computer program that calculates the ratio of signal to noise-plus-clutter as a function of target range. Probabilities of detection and false alarm are also computed. Maximum detection range can be found for a specified signal to noise-plus-clutter ratio.

The philosophy of the program to be described in this report is somewhat different from those discussed above. It is not intended to simulate the radar performance in a dynamic situation. Instead, it is intended to provide, for a specified target size, fluctuation model, and detection probability, a single number that will serve as an index of the radar's range capability -- a "figure of merit." The geophysical environment is taken into account as realistically as possible except that effects of clutter, rain, and multipath interference are omitted. The factors that are believed to be more realistically or accurately calculated than in other programs are the system noise temperature (or more specifically, the tropospheric, solar, and galactic contributions to the antenna temperature) and the tropospheric absorption loss (due to collision-broadened absorption resonances of the oxygen and water-vapor molecules). The antenna noise temperature and the atmospheric attenuation are computed directly rather than by interpolation using data entered from precalculated curves or tables; consequently, the permissible range of frequency is much greater than for most programs using precalculated temperature and absorption data.

---

*Assisted by R.G. Roll and G.T. Trotter.

The program to be described in this report computes the detection range for either a steady (nonfluctuating) target or for any of the four Swerling fluctuation cases (or for all five cases), for a specified probability of detection and a specified false-alarm probability. Fehlner's subroutine (which he named MARCUM) has been incorporated into the NRL program, with slight modifications, for this purpose. The principal modification has been to provide for calculating on the basis of false-alarm probability, rather than Marcum's false-alarm number. Another modification insures that when successive calls are made to the subroutine with the same false-alarm and number-of-pulses parameters, the bias-level calculation is not repeated. This saves an appreciable amount of computing time in the iterative procedure used to determine signal-to-noise ratio for specified probability. (Subroutine MARCUM actually does the inverse problem of computing probability for a specified signal-to-noise ratio.) Because of these and other changes, the subroutine as actually used in the NRL program has been renamed MARSWR (acronym for Marcum-Swerling); but it is basically Fehlner's MARCUM subroutine. The calculation is made assuming a square-law detector, whereas most radar receivers employ a linear-rectifier detector,* but the difference in performance of the two detector types is about 0.2 dB at most, depending on number of pulses integrated.

DATA INPUTS

The inputs to Program RGCALC for a single radar range calculation are punched on a single data card. The format specifications for this card are as follows. Each of the listed quantities is discussed briefly in the following paragraphs.

Data Item	Format Specification	Card Columns
Transmitter power, kW	F6.0	1-6
Pulse length, μsec	F6.0	7-12
Transmit antenna gain, dB	F4.0	13-16
Receive antenna gain, dB	F4.0	17-20
Target cross section, m^2	F6.0	21-26
Frequency, MHz	F6.0	27-32
Antenna ohmic loss, dB	F4.0	33-36
Receiving line loss, dB	F4.0	37-40
Transmit line loss, dB	F4.0	41-44
Antenna pattern scan loss, dB	F4.0	45-48
Miscellaneous loss, dB	F4.0	49-52
Bandwidth factor, dB	F4.0	53-56
Receiver noise factor, dB	F4.0	57-60
Number of pulses	I5	61-65
Probability of detection	F4.0	66-69
False-alarm exponent	F4.0	70-73
Swerling fluctuation case	I1	74
Target elevation angle, deg	F4.0	75-78
Galactic noise code	I2	79-80

---

*See Ref. 1 (NRL Report 6930), p. 29.

The data items in the preceding list denoted "probability of detection" (Cols. 66-69) and "false-alarm exponent" (Cols. 70-73) are actually so defined only if the "case" parameter (Col. 74) is 0, 1, 2, 3, 4, or 5. If the case parameter is 6 or 7, as will be discussed subsequently in more detail, the range is calculated for a specified signal-to-noise ratio in decibels (Case 6) or ratios (Case 7). If one signal-to-noise ratio is to be specified, it goes in card columns 66-69 in place of the probability of detection. The second signal-to-noise ratio goes in Cols. 70-73, in place of the false-alarm exponent.

Data items shown as having an F format specification can be entered as a number including a decimal point, with the number positioned anywhere within the card-column field. If an F-specification number happens to be an integer, it can also be entered without a decimal point, but in that case it must be right-adjusted within the card-column field. The specification $F6.0$ means a number of 6 characters or less including decimal point and sign, if any (positive sign is implied if no sign is given). The decimal point can be positioned anywhere in the field. Data items having an I format specification must be integers (no decimal point), and must be right-adjusted in the column field.

This single card contains all the numerical data required for a radar range calculation. However, two data cards are required for each radar calculation; the other (first) card contains any alphanumeric material that may be required to identify the radar. This material is punched anywhere in the 80-column field of the data card, and it will be printed out at the top of the page preceding the listing of input-output quantities.

Calculations for any desired number of radars can be made in one computer run by stacking the data cards in the following manner:

Cards 1, 3, 5, 7, ... Alphanumeric material identifying the radars

Cards 2, 4, 6, 8, ... Data cards giving numerical parameters corresponding to the preceding identifier cards.

If it is not desired to provide alphanumeric identifying material, blank cards should be inserted at positions 1, 3, 5, 7, . . . of the data deck. The last card of the deck is an end-of-file card. When this is encountered, the job will be terminated.

In the discussion of definitions that follows, a basic principle should be kept in mind. In any radar system, the partitioning of the system into sections called "antenna," "transmission line," "receiver," and "transmitter" is somewhat arbitrary (see NRL Report 6930, p. 47, Fig. 10, and NRL Report 7010, p. 38, Fig. 5). The points in the system at which this arbitrary partitioning is done determines the numerical values of losses, gains, power, and noise temperatures to be assigned to the factors which will subsequently be identified as L_a, L_r, L_t, P_t, G_t, G_r, T_a, T_r, and T_e ($\overline{NF}$). The range calculation will come out the same no matter how this partitioning is done if the assignment of values to all these quantities is consistent with the partitioning selected. Values of losses in decibels are to be entered on the data card as positive numbers.

Transmitter Power. See NRL Report 6930 (1), p. 11. Symbol P_t. This is the pulse power of the radar in kilowatts.

L. V. BLAKE

azimuth- and elevation-scanning radar, $L_p \approx 3.2$ dB is a reasonable assumption, although this result is based on a crude rather than a sophisticated analysis.

Miscellaneous Loss. See NRL Report 6930, p. 82-84. Symbol L_x. Among the possible losses that may be included here are collapsing loss, signal-processing loss, array-fill-time loss, beam-squint loss, polarization-rotation loss, and rain-absorption loss (if the rainstorm extent is less than the radar-to-target range). The decibel value of this loss is obtained by directly adding the decibel values of individual contributing losses.

Bandwidth Factor. See NRL Report 6930, p. 14. Symbol C_B is for the decibel loss resulting from a mismatch, in the North-filter sense, between the pulse characteristics and the receiver filter transfer characteristic. For a simple pulse radar, this relationship can be analyzed adequately in terms of the pulse length and shape, and the filter bandwidth. For most radars of this type it is reasonable to assume $C_B = 0$ dB, in the absence of specific knowledge to the contrary. For pulse-compression radars, there is usually some loss associated with the compression filter, ranging from perhaps 0.5 dB to several decibels, depending on the technique employed and the compression ratio. As is done with the loss factors, C_B is to be entered as a positive decibel number.

Receiver Noise Factor. (Also called receiver noise figure; although "figure" is perhaps the most common usage, IEEE standards give preference to "factor.") See NRL Report 6930, p. 50. Symbol NF or F_n. The receiver noise factor and receiver noise temperature are alternative ways of expressing the same property of the receiver, but the noise factor has been chosen here because it is the quantity more commonly given in receiver specifications. The decibel value of the noise factor is to be entered on the data card.

Number of Pulses. See NRL Report 6930, pp. 71 and 72. Symbol M. If this number is determined by a signal processor, it must be found by reference to the characteristics of the processor. When it is determined by the scanning action of the radar antenna, and if a simple azimuth scan is employed, the appropriate formula is

$$M = \frac{\phi \cdot PRF}{6 \cdot RPM \cdot \cos \theta_e},$$ (1)

where ϕ is the azimuthal half-power beamwidth, degrees; PRF is the pulse repetition frequency, hertz; RPM is the rotation rate of the antenna, revolutions per minute; and θ_e is the elevation angle of the target. (The term $\cos \theta_e$ is significant only when a target is at an elevation angle of about 10 degrees or more. For vertical-fan-beam radars the range is usually calculated at an elevation angle low enough so that $\cos \theta_e \approx 1$.)

For simultaneous azimuth- and elevation-scanning radars, assuming a sawtooth-motion elevation scan and a uniform-speed-rotation azimuth scan, the appropriate formula is

$$M = \frac{\phi \cdot \theta \cdot PRF}{6 \, \omega_e \cdot t_v \cdot RPM \cdot \cos \theta_e},$$ (2)

in which ϕ, PRF, RPM, and θ_e have the same definitions as before, θ is the vertical beamwidth, ω_e is the vertical scanning speed in degrees per second at the target elevation angle, and t_v is the vertical-scan period in seconds, including the dead time if any.

The number of pulses to be used for radars of other scan types must be analyzed on an individual basis, as discussed in NRL Report 6930.

The number of pulses is entered on the data card as an integer (I-format). Consequently the number must be right-adjusted in the field of columns 61 through 65. If the number calculated from the above formulas is not an integer, it should be rounded off to the nearest integer. (The subroutine that calculates detection probability requires an integer for the number of pulses integrated.)

Probability of Detection. See NRL Report 6930, pp. 18 and 19. Symbol P_d. Probability is here given in the mathematical sense of a number between 0 and 1 (not as a percentage figure). Values larger than 0.99 should not be entered because computational difficulties result. Also, values smaller than the false-alarm probability P_{fa} are meaningless; for practical purposes, P_d should be at least an order of magnitude larger than P_{fa}. (Ordinarily it is many orders of magnitude larger.) Typical values of P_d of interest range from about 0.1 to 0.95.

False-Alarm Exponent. See NRL Report 6930, pp. 18-19 Symbol $-\log_{10} P_{fa}$. Typical values of false-alarm probability range from 10^{-4} to 10^{-12}. The number to be entered on the data card is the positive value of the exponent (power of ten). Thus, for $P_{fa} = 10^{-6}$, enter the number 6.0 on the data card: for $P_{fa} = 2.5 \times 10^{-6}$, enter 5.6.

Swerling Fluctuation Case. See NRL Report 6930, p. 28. Five cases are considered, with 0 representing the nonfluctuating target and integers 1 through 4 representing the 4 Swerling cases as defined in NRL Report 6930 (and elsewhere). If the numbers 0, 1, 2, 3, or 4 are punched on the data card in Column 74, the corresponding fluctuation case will be calculated. If 5 is punched, all 5 cases will be calculated.

Further options are provided by a 6 or a 7 punch in Column 74. A 6 punch signifies that the range is to be calculated for a specified signal-to-noise ratio rather than for specified probabilities of detection and false alarm. For this case, the signal-to-noise ratio, in decibels, is punched in Columns 66-69, where probability of detection would ordinarily appear. If a 7 is punched in Column 74, the calculation of range will be made for two different signal-to-noise ratios, one given in Columns 66-69, the other in Columns 70-73. When either 6 or 7 is punched in Column 74, the number-of-pulses entry, Columns 61-65, is ignored. Likewise, if 6 is punched, the false-alarm exponent entry, Columns 70-73, is ignored.

Target Elevation Angle. See NRL Report 6930, pp. 48, 68, 69, and 72 through 80. As mentioned in the Introduction, this factor enters into the "quasi-free-space" range calculation because the effect of the earth's atmosphere on the antenna noise temperature and on absorption loss is taken into account. The elevation angle is to be entered in degrees. If a range calculation applicable in empty space is desired, a close approximation can be obtained, except at frequencies near the water vapor and near or above the oxygen resonances (22 and 60 GHz, respectively) by setting the elevation angle to 90 degrees, because for this setting the absorption is virtually negligible. Also, since the calculated absorption is one of the printed-out quantities, correction for it can be made.

Galactic Noise Code. See NRL Report 6930, p. 49, Fig. 11. As shown in the referenced figure, the noise received from the galaxy to which the solar system belongs varies depending on the part of the galaxy toward which the antenna is pointed. This direction is not

usually predictable. Therefore the options of calculating the radar range for three choices of galactic and solar noise levels are provided. The codes are 1 for minimum galactic noise, 0 for average noise, and +1 for maximum galactic noise. (The maximum and minimum values are shown by dashed lines in the referenced figure of NRL Report 6930.) The number entered is to be right-adjusted in Columns 79-80

PROGRAM OUTPUT

The output of Program RGCALC is a single printed page for each set of data inputs (two data cards). The alphanumeric material of the first data card is printed at the top of the page. Then the numerical input data are printed, both as a record of the data and to ensure that the data card was correctly punched. Next are printed some intermediate output quantities such as the computed noise temperatures of the system components and of the overall receiving system, and the tropospheric absorption for a two-way path through the entire troposphere at the specified elevation angle. Then, if the number punched in Column 74 of the numerical data input card was 5 or less, the calculated range or ranges are printed for the Swerling case or cases specified. Along with each range figure are also given the calculated signal-to-noise ratio in decibels and the tropospheric attenuation for that range.

If the "case" parameter of Column 74 is 6 or 7, the printed output is modified slightly to reflect the fact that the range has been calculated on the basis of an assumed signal-to-noise ratio rather than for specific probabilities of detection and false alarm and a specific fluctuation model.

Figures 1 through 6 are illustrations of the input data cards and resultant printed output for three different "case" options, namely 1, 5, and 7. Cases 0, 2, 3, and 4 produce output results similar to that shown for Case 1, and Case 6 produces an output similar to that of Case 7, except that the range is then calculated for only one signal-to-noise ratio. The radar parameters of these sample calculations are fictitious.

EQUATIONS AND ALGORITHMS

It has been mentioned that Eq. (12) of NRL Report 6930 (1) is the basis of Program RGCALC. The equation is

$$R_{max} = 129.2 \left[\frac{P_{t(kW)} \tau_{\mu sec} G_t G_{r0} F_t^2 F_r^2}{f_{MHz}^2 T_s V_o C_B L} \right]^{1/4} . \tag{3}$$

The symbols in this equation have been previously defined in this report except for those that follow:

F_t, F_r — pattern-propagation factors for the radar transmitter-to-target and target-to-receiver paths, respectively. In Program RGCALC, $F_t = F_r = 1$.

T_s — the receiving system noise temperature, kelvins

V_o — the visibility factor, or predetection signal-to-noise ratio required for the specified probability of detection of the target echo

L -- the system loss factor, equal to the product of all the component loss factors.* (In decibels, L is the sum of the component decibel losses.)

Program RGCALC is an executive routine which reads the data cards and then transmits the data to a subroutine named RANGE. This subroutine calls other subroutines which calculate the system noise temperature T_s, the signal-to-noise ratio V_o, and a table

Fig. 1 -- Data cards for calculating range of a UHF search radar, Swerling case 1 target fluctuation (1 punch in Col. 74 of second card).

---

*Loss factor is sometimes expressed as a number less than one, in which case it would belong in the numerator of the range equation. Here the correct engineering definition of loss as the reciprocal of gain is employed, so that $L \geq 1$.

```
RADAR NAME OR DESCRIPTION --

FICTITIOUS UHF SEARCH RADAR


RADAR AND TARGET PARAMETERS (INPUTS) --

PULSE POWER, KW ................................          500.0
PULSE LENGTH, MICROSEC .........................      20.0000
TRANSMIT ANTENNA GAIN, DB ......................         22.0
RECEIVE ANTENNA GAIN, DB .......................         22.0
FREQUENCY, MHZ .................................        315.0
RECEIVER NOISE FACTOR (FIGURE), DB .............          2.5
BANDWIDTH CORRECTION FACTOR, DB ................          0.5
ANTENNA OHMIC LOSS, DB .........................          0.0
TRANSMIT TRANSMISSION LINE LOSS, DB ............          0.3
RECEIVE  TRANSMISSION LINE LOSS, DG ............          0.5
SCANNING-ANTENNA PATTERN LOSS, DB ..............          1.6
MISCELLANEOUS LOSS, DB .........................          0.0
NUMBER OF PULSES INTEGRATED ....................           45
PROBABILITY OF DETECTION .......................        0.500
FALSE-ALARM PROBABILITY, NEGATIVE POWER OF TEN           8.0
TARGET CROSS SECTION, SQUARE METERS ............       5.0000
TARGET ELEVATION ANGLE, DEGREES ................         0.50
AVERAGE SOLAR AND GALACTIC NOISE ASSUMED
PATTERN-PROPAGATION FACTORS ASSUMED = 1.


        ********************************************


CALCULATED QUANTITIES (OUTPUTS) --

NOISE TEMPERATURES, DEGREES KELVIN --
        ANTENNA (TA) ...........................        251.6
        RECEIVING TRANSMISSION LINE (TR) .......         35.4
        RECEIVER (TE) ..........................        225.7
        TE X LINE-LOSS FACTOR = TEI ............        253.2
        SYSTEM (TA + TR + TEI) .................        540.2
TWO-WAY ATTENUATION THROUGH ENTIRE TROPOSPHERE, DB        1.0
```

SWERLING FLUCTUATION CASE	SIGNAL-TO-NOISE RATIO, DB	TROPOSPHERIC ATTENUATION, DECIBELS	RANGE, NAUTICAL MILES
1	1.90	0.97	209.9

Fig. 2 -- Program RGCALC output for data cards of Fig. 1
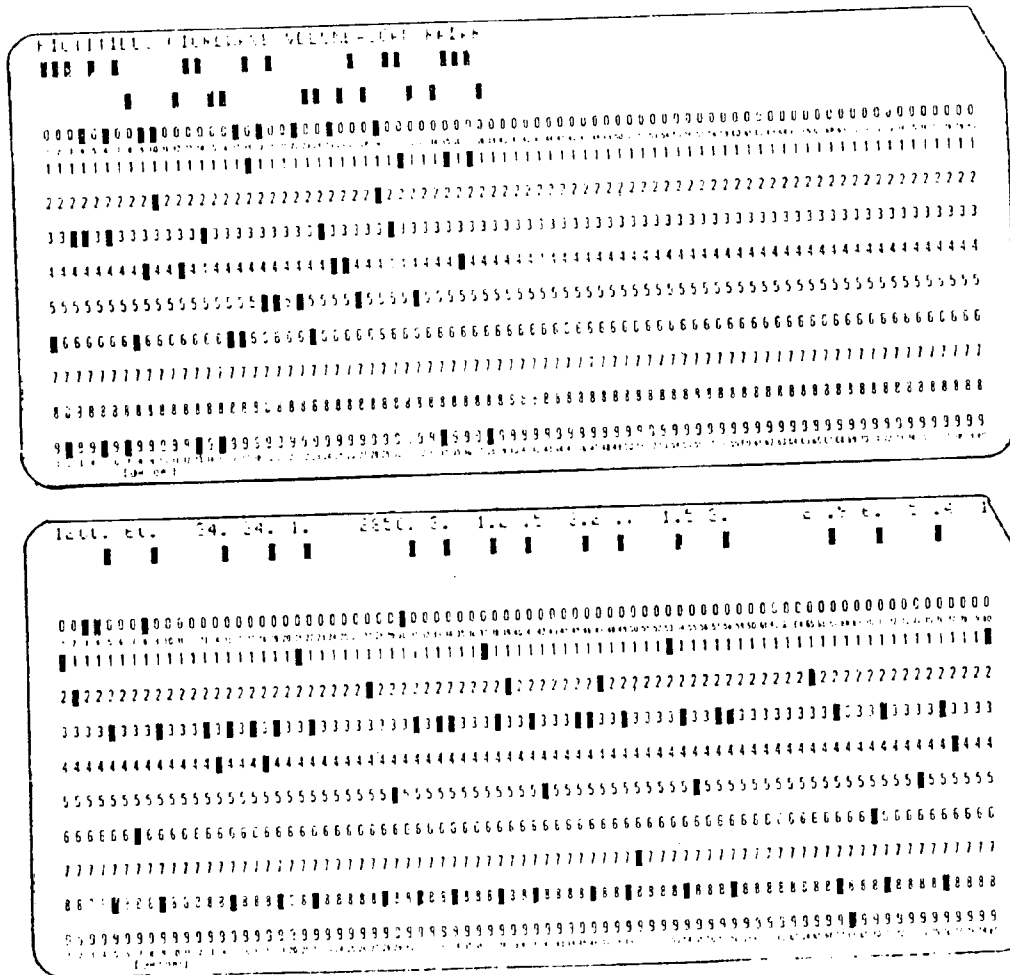
Fig. 3 — Data cards for calculating range of a microwave volume-scanning radar, Swerling cases 0, 1, 2, 3, 4 fluctuation (5 punch in Col. 74 of second card)

(array) of tropospheric absorption losses in decibels at range increments along the (refracted) ray path for the specified elevation angle. Then a system loss factor L, with tropospheric absorption loss omitted, is calculated. The range equation arithmetic is then performed, using the input data and the calculated values of T_s, V_o, and L. Then, in the table of calculated absorption loss values, a value of this loss corresponding to the calculated range is found by interpolation. The range is corrected by a factor corresponding to this loss factor; then the new loss factor corresponding to this corrected range is found, again by interpolation; this iteration is repeated until the last correction corresponds to less than 0.1 dB, in a subroutine named ITERAT.

14   L. V. BLAKE

RADAR NAME OR DESCRIPTION =,

FICTITIOUS MICROWAVE VOLUME-SCAN RADAR

RADAR AND TARGET PARAMETERS (INPUTS) --

```
PULSE POWER, KW ,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,      1200,0
PULSE LENGTH, MICROSEC ,,,,,,,,,,,,,,,,,,,,,,,,,,,,,       60,0000
TRANSMIT ANTENNA GAIN, DB ,,,,,,,,,,,,,,,,,,,,,,,,,,         34,0
RECEIVE ANTENNA GAIN, DB ,,,,,,,,,,,,,,,,,,,,,,,,,,,         34,0
FREQUENCY, MHZ ,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,        2850,0
RECEIVER NOISE FACTOR (FIGURE), DB ,,,,,,,,,,,,,,             3,0
BANDWIDTH CORRECTION FACTOR, DB ,,,,,,,,,,,,,,,,,            1,5
ANTENNA OHMIC LOSS, DB ,,,,,,,,,,,,,,,,,,,,,,,,,,,           3,0
TRANSMIT TRANSMISSION LINE LOSS, DB ,,,,,,,,,,,,,,          0,5
RECEIVE  TRANSMISSION LINE LOSS, DB ,,,,,,,,,,,,,,          1,2
SCANNING=ANTENNA PATTERN LOSS, DB ,,,,,,,,,,,,,,            3,2
MISCELLANEOUS LOSS, DB ,,,,,,,,,,,,,,,,,,,,,,,,,,           0,7
NUMBER OF PULSES INTEGRATED ,,,,,,,,,,,,,,,,,,,,,             2
PROBABILITY OF DETECTION ,,,,,,,,,,,,,,,,,,,,,,,,,         0,900
FALSE=ALARM PROBABILITY, NEGATIVE POWER OF TEN               6,0
TARGET CROSS SECTION, SQUARE METERS ,,,,,,,,,,,,,         1,0000
TARGET ELEVATION ANGLE, DEGREES ,,,,,,,,,,,,,,,,            0,40
MAXIMUM SOLAR AND GALACTIC NOISE ASSUMED
PATTERN=PROPAGATION FACTORS ASSUMED = 1,
```

                •••••••••••••••••••••••••••••••••••••

CALCULATED QUANTITIES (OUTPUTS) --

NOISE TEMPERATURES, DEGREES KELVIN =,
```
        ANTENNA (TA) ,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,       211,1
        RECEIVING TRANSMISSION LINE (TR) ,,,,,,,,          90,3
        RECEIVER (TE) ,,,,,,,,,,,,,,,,,,,,,,,,,,,,,        288,6
        TE X LINE=LOSS FACTOR = TEI ,,,,,,,,,,,,,          380,5
        SYSTEM (TA + TR + TEI) ,,,,,,,,,,,,,,,,,,,         683,9
TWO-WAY ATTENUATION THROUGH ENTIRE TROPOSPHERE, DB           3,6
```

SWERLING FLUCTUATION CASE	SIGNAL-TO-NOISE RATIO, DB	TROPOSPHERIC ATTENUATION, DECIBELS	RANGE, NAUTICAL MILES
0	10,65	2,88	126,9
1	18,69	2,17	83,2
2	14,83	2,52	101,9
3	14,83	2,52	101,9
4	12,86	2,69	113,0

Fig. 4 -- Program RGCALC output for data cards of Fig. 3
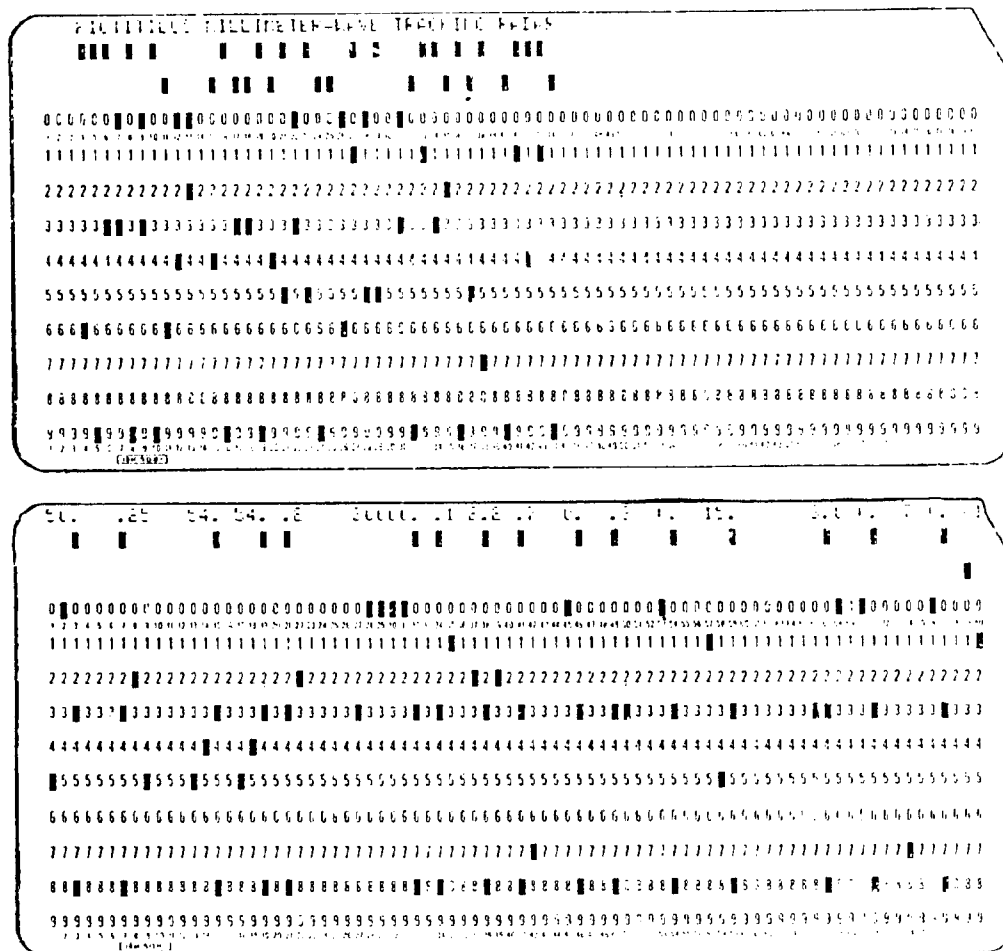
Fig. 5 — Data cards for calculating range of a millimeter-wave tracking radar on a specified signal-to-noise-ratio basis (7 punch in Col. 74 of second card)

16   L. V. BLAKE

RADAR NAME OR DESCRIPTION =-

FICTITIOUS MILLIMETER-WAVE TRACKING RADAR

RADAR AND TARGET PARAMETERS (INPUTS) --

PULSE POWER, KW 50.0
PULSE LENGTH, MICROSEC 0.2500
TRANSMIT ANTENNA GAIN, DB 54.0
RECEIVE ANTENNA GAIN, DB 54.0
FREQUENCY, MHZ 30000.0
RECEIVER NOISE FACTOR (FIGURE), DB 15.0
BANDWIDTH CORRECTION FACTOR, DB 0.0
ANTENNA OHMIC LOSS, DB 0.1
TRANSMIT TRANSMISSION LINE LOSS, DB 0.7
RECEIVE TRANSMISSION LINE LOSS, DB 2.2
SCANNING-ANTENNA PATTERN LOSS, DB 0.0
MISCELLANEOUS LOSS, DB 0.3
SIGNAL-TO-NOISE RATIO, DB 3.0
SIGNAL-TO-NOISE RATIO, DB 0.0
TARGET CROSS SECTION, SQUARE METERS 0.2000
TARGET ELEVATION ANGLE, DEGREES 0.00
MINIMUM SOLAR AND GALACTIC NOISE ASSUMED
PATTERN-PROPAGATION FACTORS ASSUMED = 1.

****

CALCULATED QUANTITIES (OUTPUTS) --

NOISE TEMPERATURES, DEGREES KELVIN --
          ANTENNA (TA) 282.6
          RECEIVING TRANSMISSION LINE (TR) 191.3
          RECEIVER (TE) 8880.6
          TE X LINE-LOSS FACTOR = TEI 14738.1
          SYSTEM (TA + TR + TEI) 15212.0
TWO-WAY ATTENUATION THROUGH ENTIRE TROPOSPHERE, DB   34.4

RANGE =   22.1 N. MI., TROPOSPHERIC ATTENUATION =   7.27 DB

FOR SPECIFIED SIGNAL-TO-NOISE RATIO =   3.00 DB

RANGE =   25.0 N. MI., TROPOSPHERIC ATTENUATION =   8.15 DB

FOR SPECIFIED SIGNAL-TO-NOISE RATIO =   0.00 DB

Fig. 6 — Program RGCALC output for data cards of Fig. 5

Figure 7 is a flow chart showing the sequence of events in the computation. As indicated, checks are made at several points which permit certain portions of the calculation to be omitted if more than one range calculation is being made and if results of some steps in the preceding calculation can be used.

In the following sections, the algorithms of the various subroutines of the program will be described. The names in parentheses following the subroutine name are its parameters in the calling sequence.

Subroutine ALPHA (FMHZ)

This subroutine performs the initial step in the computation of tropospheric absorption loss and noise temperature; it computes a set of absorption coefficients in decibels per nautical mile for a set of altitudes above sea level from 0 to 100,000 ft. The first 21 of these altitudes (from 0 to 2000 ft) are at intervals of 100 ft; the next 28, to 30,000 ft, are at intervals of 1000 ft; the next 20, to 70,000 ft, are at intervals of 2000 ft; and the last 6, to 100,000 ft, are at intervals of 5000 ft. This graduation of height intervals reflects the fact that the absorption coefficient changes more rapidly in the lower atmosphere than it does at higher altitudes.

The absorption coefficients are calculated at each of these altitudes for both oxygen and water vapor, and the two coefficients are added to obtain a total absorption coefficient. The resulting array of 75 coefficients, for the frequency FMHZ (first parameter of the calling sequence) is named ALPH (J, 75), J = 3. It is transmitted as output via a COMMON block named PTR.

The computation is done using the theory of Van Vleck as described in NRL Report 7010 (1), except that in the previous version of the subroutine described there, Van Vleck's centroid approximation was used, and it is not valid in the region near the oxygen resonances from about 50 to 70 GHz. The new version of the subroutine, as now used in Program RGCALC, performs a more exact calculation by summing the separate absorption contributions of each of 46 individual oxygen resonance frequencies. Consequently, range calculations can now be made within as well as outside the frequency region 50 to 70 GHz.

The calculations are made for the standard dry atmosphere known as the U.S. Extension to the International Civil Aviation Organization (ICAO) Standard Atmosphere (10). The model for water-vapor content of the atmosphere is based on a humidity profile given by Sissenwine and others (11) as representative of the midlatitude mean humidity. This midlatitude mean, however, has a surface water-vapor content of 5.947 gm/m^3, and it was desired (in accordance with prevailing practice) to adopt the surface water-vapor density value of 7.5 gm/m^3 for the absorption computations. Therefore, the values of the Sissenwine model were all multiplied by the factor 7.5/5.947 = 1.261. The tabulation given in Sissenwine's report is for altitudes at intervals of 2 km in the region of interest here. The values corresponding to the altitudes specified in Subroutine ALPHA were obtained by means of an interpolation technique developed for digital-computer plotting of a smooth curve through a set of data points (12).

The sets of 75 values of pressure, temperature, and water-vapor density values defining this model atmosphere are entered into Subroutine ALPHA in the form of Fortran DATA statements (arrays PP, TT, and RR), thus obviating any necessity of reading them in from
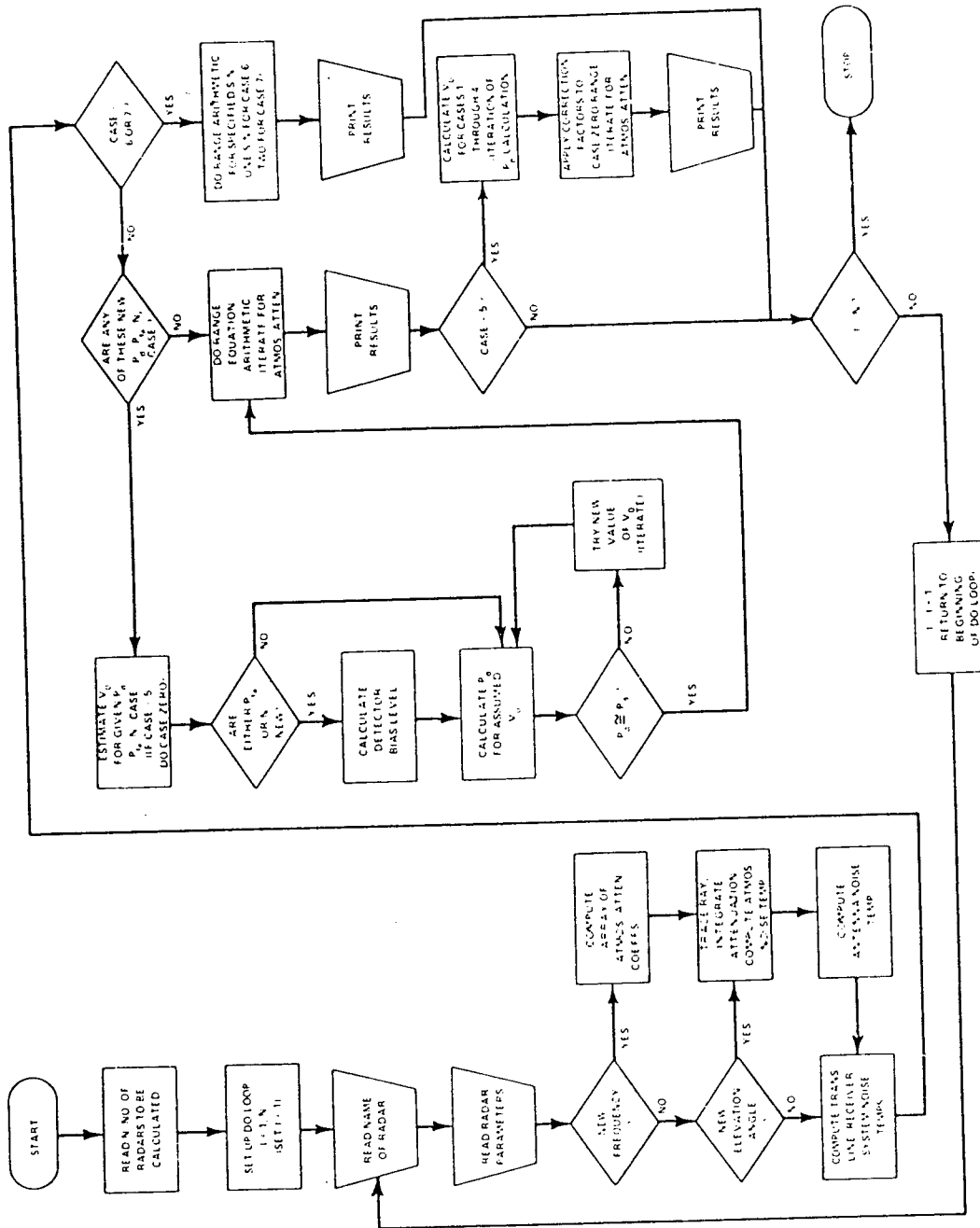
Fig. 7 — Flow chart for Program RGCALC

a separate deck of data cards each time the program is run. The set of 46 oxygen resonance frequencies is similarly entered via DATA statements; these frequencies are separated into two arrays, FTRP and FTRM, corresponding to two classes of quantum-mechanical state transitions of the oxygen molecule. The details of the calculations are described by Meeks and Lilley (13); their formulations of the oxygen absorption equations were employed. (The frequencies FTRP correspond to their symbol ν_{N+}, and FTRM corresponds to ν_{N-}.)

The only deviation from the Meeks and Lilley calculations was the use of a slightly different dependence of line width on altitude. The model of Reber, Mitchell, and Carter (14) was used for this part of the calculation.

Subroutine ATLØSS (FMHZ, ELEV, ATMP)

The input parameters are FMHZ, frequency in megahertz, and ELEV, elevation angle in degrees. The output parameter ATMP is the tropospheric noise temperature computed for the specified frequency and elevation angle. Another output, transmitted via COMMON block RGA, is an array of absorption values named ATTN (decibels) corresponding to a set of range values along the ray path at angle ELEV, corresponding to the altitude values of Subroutine ALPHA. The corresponding array of range values RG is similarly transmitted as output.

Subroutine ALPHA is called by Subroutine ATLØSS, and the resulting array of 75 values of absorption coefficients ALPH(J,75), J = 3, is used to calculate cumulative absorption along the ray path at angle ELEV. The ray path in the refracting atmosphere is computed by numerical-integration ray tracing, from the formula (15)

$$R(h_1) = \int_0^{h_1} \left(\frac{dR}{dh}\right) dh \tag{4}$$

in which

$$\frac{dR}{dh} = \frac{n(h)}{\sqrt{1 - \left[\frac{n_0 \cos \theta_0}{n(1-h/r_0)}\right]^2}} \cdot \tag{5}$$

Here R is the radar range corresponding to height h_1 as measured along the ray path of the initial elevation angle θ_0, $n(h)$ is the refractive index height profile, n_0 is the value of n at $h = 0$, and r_0 is the radius of the earth (more specifically, it is the distance from the earth's center to the initial point of the ray).

The attenuation is then computed along this ray path by numerically integrating the equation

$$A(R_1) = 2 \int_0^{h_1} \gamma(h) \cdot \frac{ds}{dh} \cdot dh, \tag{6}$$

L. V. BLAKE

in which $R_1 = R(h_1)$, and $\gamma(h)$ is the absorption coefficient at height h, as given by the array ALPH (J, 75), J = 3. The derivative ds/dh is equal to $[1/n(h)] \cdot dR/dh$; that is, R is the radar range measured along the ray path, and s is the geometric distance along the same path. The derivatives ds/dh and dR/dh are computed in a short subroutine named $DDH(H)$, in which the single parameter is the height H. The derivatives are named $DSDH$ and $DRDH$ and are transmitted to Subroutine ATLOSS via a COMMON block named DRS.

The refractive index model used is given by (2)

$$n(h) = 1 + 0.000313 \, e^{-kh} \tag{7}$$

where $k = 4.3848 \times 10^{-5}$ if h is in feet. The earth's radius is assumed to be 6370 km, or 2.0899×10^7 ft.

A special technique is used to perform the integrations in the vicinity of $h = 0$ for the special case $\theta_0 = 0$, because in that case dR/dh and ds/dh both become infinite. This technique was described by the author in a paper published in 1968 (15).

The tropospheric noise temperature T_{atm} (Fortran name ATMP) is computed by numerical integration of the equation

$$T_{atm} = 0.2303 \int_0^\infty \gamma(R) \, T_t(R) \exp\left[-0.2303 \int_0^R \gamma(r) \, dr \right] dR, \tag{8}$$

where dR is taken along the ray path. T_t is the thermal temperature of the troposphere; its values are transmitted to Subroutine ATLOSS from ALPHA via COMMON block TMP.

The previously described modification of Simpson's rule *cannot* be used to perform this integration because it is an integration with respect to R (range) rather than h (height). The h intervals, as described in the section on Subroutine ALPHA, are uniform (over each of the four height regions); however, the corresponding R intervals are not uniform. Another special modification of Simpson's rule was devised to handle this problem; it is embodied in Subroutine INTGRT, which is called by ATLOSS to perform the numerical integration of Eq. (8). Further details of the absorption and noise temperature calculations will be given in a report to be written in the near future, in which curves for absorption and noise temperature as functions of frequency and elevation angle will be presented.

Subroutine TEMP (FMHZ, ELEV, K, ANF, ALA, ALR, ATMP,
TA, TR, TE, TEI, TSYS)

The input parameters FMHZ, ELEV, and ATMP are the same as those of Subroutine ATLOSS. The other input parameters are

K — galactic noise code (−1, 0, +1)

ANF — receiver noise factor $\overline{NF}$, dB

ALA — antenna loss factor L_a dB

ALR — receiving line loss factor L_r, dB.

The output parameters are

    TA — antenna noise temperature T_a, kelvins

    TR — receiving-transmission-line noise temperature T_r

    TE — receiver noise temperature T_e

    TEI — product of T_e and L_r

TSYS — system noise temperature T_s.

The antenna noise temperature is computed by use of equations given in NRL Report 6930, p. 49, and Report 7010, pp. 40 through 44 (1). The sky temperature, named TAI, is first computed from the equation

$$T_{sky} = (T_{gal} + T_{bb})/L_{atm} + T_{a(sun)} + T_{atm} \tag{9}$$

in which T_{gal} is the galactic noise temperature, T_{bb} is the cosmic blackbody temperature (2.7 K), L_{atm} is the atmospheric loss factor (expressed as a power ratio $\geqslant 1$), $T_{a(sun)}$ is the solar contribution to the antenna temperature (assuming that the sun is in an average-level sidelobe of the antenna pattern), and T_{atm} is the atmospheric noise temperature (ATMP, obtained from Subroutine ATLØSS). The galactic temperature is given by

$$T_{gal} = T_{100} \cdot (100/f_{MHz})^{2.5}. \tag{10}$$

The quantity T_{100} is the galactic temperature at the frequency $f_{MHz} = 100$ MHz. Its numerical value depends on the galactic noise code K according to the following prescription.

K	T_{100} (kelvins)
−1	500
0	3050
+1	18,650

The solar contribution to antenna temperature $T_{a(sun)}$ is obtained from a table of values of the solar noise temperature T_{sun} entered via a DATA statement; the table corresponds to frequencies in the range 100 MHz to 10 GHz, and the values are taken from Fig. 6 of Report 7010, p. 43 (1). At frequencies between the tabulated values, T_{sun} is found by linear interpolation. Above 10 GHz, T_{sun} is assumed to have the constant value 10,000 kelvins. The solar temperature T_{sun} is related to $T_{a(sun)}$ by the equation

$$T_{a(sun)} = T_{sun} \times 4.75 \times 10^{-5}/L_{atm}. \tag{11}$$

The numerical factor takes into account the assumed unity-gain average sidelobe level, the ratio of the sun's noise diameter to the total solid angle (4π steradians) viewed by the antenna including its side and back lobes, and the assumption that the sun is on the average ten times noisier than indicated by the referenced curve, which portrays the "quiet sun." Then, $T_{a(sun)}$ is decreased by a factor of 10 if K = −1 and increased by 10 if K = +1, where K is the galactic (and solar) noise code.

L. V. BLAKE

The resulting value of sky temperature TAI is multiplied by 0.876 to take into account the fraction of the total antenna pattern subtended by the sky, and to this is added the contribution due to antenna-loss noise, in accordance with Eq. (37) of NRL Report 6930.

The transmission line and receiver noise temperatures T_r and T_e are computed in accordance with Eqs. (40) and (41) of NRL Report 6930, p. 50, and combined to give the system noise temperature by the equation

$$T_s = T_a + T_r + L_r T_e,\tag{12}$$

where L_r is the receiving line loss factor. The product $L_r T_e$ is also reported to Subroutine RANGE as the parameter TEI, and is printed as an intermediate output of the program along with T_a, T_r, T_e, and T_s.

Subroutine PDSN (PDT, PFA, NPULS, KASE, SDB)

The purpose of this subroutine is to find the signal-to-noise ratio required for detection, SDB, for the specified probability of detection PDT, false-alarm probability PFA (expressed as a positive number representing the negative power of ten), number of pulses integrated NPULS, and Swerling fluctuation case, KASE. Subroutine PDSN does not perform the calculation; it merely manages it by calling other subroutines. The actual calculation requires an iteration, which it performed by Subroutine INVERS, called by PDSN. Before calling INVERS, PDSN estimates a range of decibel values (lower value DB1, upper value DB2) likely to contain the true value SDB. An empirical formula is used for this purpose. This procedure minimizes the number of iterations required. Subroutine PDSN is called from Subroutine RANGE, and the resultant value of SDB is used as a factor in the range calculation. When the "case" parameter of Subroutine RANGE is 6 or 7 (Col. 74 of the data card), PDSN is not called, since the signal-to-noise ratio is then a direct input and need not be calculated.

Subroutine INVERS (XMIN, XMAX, XLØ, XHI, NSIG, LIM, NØI, X, Fl, FT, F)

This subroutine performs an iteration to determine the value of the argument X of a function F(X) which will, within a specified accuracy, cause F(X) to equal FT, a specified value of the function. The accuracy parameter is NSIG — the number of significant figures to which agreement is desired between F(X) and FT. LIM specifies a limit on the number of iterations permitted, and NØI (output parameter) reports the number of iterations actually performed. Fl is the actual final value of F. It is required that F be a monotonic function of X within the permissible range of variation of X, which is from XMIN to XMAX, and that the value FT exists within this range. The input parameters XIØ and XHI define a region in which it is guessed that the desired value of X will be found. If no knowledge exists by which to estimate this region, XLØ and XHI can be set equal to XMIN and XMAX; however, the more narrowly the region is defined, the fewer will be the iterations required.

If the slope dF/dX becomes nearly zero in some part of the range from XMIN to XMAX, and if this region is contained between XLØ and XHI, the iteration will converge very slowly, or conceivably not at all. It is for this reason, as mentioned earlier, that

values of probability of detection greater than 0.99 should not be specified. Evidently Subroutine MARSWR, which is involved in the iteration, does not define well the slope of the "function" in this region.

Function PD (SNDB)

In order to define a function on which INVERS can operate, the Fortran FUNCTION PD is used, with signal-to-noise ratio (dB) as the argument SNDB. This function merely calls Subroutine MARSWR, which calculates the probability of detection.

Subroutine MARSWR (SNDB, N, FA, KASE, PN)

As has been discussed, this subroutine is basically the subroutine of Fehlner (6) which he named MARCUM. It was renamed MARSWR after a few changes in it were made to adapt it to the requirements of Program RGCALC. The input parameters are SNDB, the signal-to-noise ratio in decibels; N, the number of pulses integrated; FA, the false-alarm exponent; and KASE, the Swerling fluctuation case.

As has been mentioned, Fehlner's original subroutine calculates the probability of detection PN for a specified value of Marcum's false-alarm number rather than on the basis of false-alarm probability. The power-of-ten exponent of the false-alarm number is named FAN in the subroutine. The relation between FAN and the false-alarm probability, P_{fa} is

$$FAN = \log_{10}\left[\frac{\log_e 0.5}{\log_e (1 - P_{fa})}\right].$$ (13)

This relationship is used in Subroutine MARSWR to convert the input parameter FA to the internal parameter FAN.

In using Subroutine MARCUM, it was found that an appreciable portion of the computing time is spent in computing the bias level YB. If successive calls to MARCUM are made with the same values of N and FA (but with different values of SNDB and KASE), it is not necessary to repeat the bias-level calculation. Therefore a provision for omitting that part of the calculation, when successive calls to MARSWR are made with the same values of N and FA, has been added to the subroutine.

Functions named DGAM, DEVAL, GAM, and SUMLOG are part of the MARCUM subroutine, which was originally written in Fortran II for use with an IBM computer. These functions are also incorporated into MARSWR. The only changes made in them were those necessary to adapt them for use on the NRL CDC-3800 computer. (Some of these adapting changes were made by Stanley Gontarek, of the Naval Air Systems Command.) A further slight change was made in Function SUMLOG; the array named A therein was given a dimension 1000 rather than 200 as in the original MARCUM subroutine. This increased dimensioning saves computing time if successive calls to SUMLOG are made with values of N greater than 200.

FORTRAN PROGRAM LISTING

The Fortran program, subroutines, and functions are listed on the following pages. The names of the listed routines and their computer lengths (number of locations required) are as follows:

Name	Locations Required	
	Octal	Decimal
Program RGCALC	232	154
Subroutine RANGE	2030	1048
Subroutine TEMP	362	242
Subroutine ATLØSS	756	494
Subroutine DDH	112	74
Subroutine ALPHA	520	336
Subroutine INTGRT	211	137
Subroutine ITERAT	220	144
Subroutine PDSN	230	152
Function PD	63	51
Subroutine INVERS	714	460
Subroutine MARSWR	3123	1619
Function DGAM	216	142
Function DEVAL	106	70
Function GAM	206	134
Function EVAL	111	73
Function SUMLØG	4203	2179
Totals	16525	7509

06/20/72

```
      SUBROUTINE RANGE(PTKW,TAUMS,GTDB,GRDB,SIGSM,FMHZ,ALADB,ALRDB,
    1   ALTDB,ALPDB,ALXDB,CBDB,ANFDB,NPULS,PD,PFA,KASE,ELEV,NOISE)
      COMMON/RGA/ RG(75), ATTN(3,75)
      IF (KASE ,EQ. 5) 10,11
10    IKASE = 0
      GO TO 12
11    IKASE = KASE
12    CALL TEMP(FMHZ,ELEV,NOISE,ANFDB,ALADB,ALRDB,ATMP,TA,TR,TE,TEI,
    1     TSYS)
      IF (KASE ,GE. 6) 60,61
60    SNDB=PD
      GO TO 62
61    CALL PDSN(PD,PFA,NPULS,IKASE,SNDB)
62    FACDB=GTDB+GRDB+CBDB+ALTDB-ALPDB+ALXDB-SNDB
      FAC=10.**(FACDB*.1)
      RNGO=129.2*(PTKW*TAUMS*SIGSM*FAC/ (FMHZ*FMHZ*TSYS))**.25
      RNGA=RNGO
      CALL ITERAT(RNGA,ATT)
      PRINT 1
      PRINT 100, PTKW
      PRINT 101, TAUMS
      PRINT 102, GTDB
      PRINT 103, GRDB
      PRINT 104, FMHZ
      PRINT 105, ANFDB
      PRINT 106, CBDB
      PRINT 108, ALADB
      PRINT 109, ALTDB
      PRINT 110, ALRDB
      PRINT 111, ALPDB
      PRINT 112, ALXDB
      IF (KASE ,GE. 6) 63,64
63    PRINT 117, SNDB
      IF (KASE ,EQ. 7) PRINT 117, PFA
      GO TO 65
64    PRINT 107, NPULS
      PRINT 113, PD
      PRINT 114, PFA
65    PRINT 115, SIGSM
      PRINT 116, ELEV
      IF (NOISE) 50,51,52
50    PRINT 55
      GO TO 53
51    PRINT 56
      GO TO 53
52    PRINT 57
53    PRINT 58
      PRINT 60
      PRINT 2
      PRINT 120
      PRINT 121, TA
      PRINT 122, TR
      PRINT 123, TE
      PRINT 1123, TEI
      PRINT 124, TSYS
      PRINT 125, ATTN(3,75)
```

L. V. BLAKE

05/12/72

```
   PROGRAM RGCALC
   DIMENSION NAME(10)
 2 READ 3, NAME
   IF (EOF,60) 10,11
10 STOP
11 PRINT 22
22 FORMAT(15X,*RADAR NAME OR DESCRIPTION -- */)
   PRINT 30, NAME
   READ 4,      PT,TAU,GT,GR,SIG,FM,ALA,ALR,ALT,ALP,ALX,CB,ANF,NP,PD,
 1    FA,KA,EL,NS
   CALL RANGE(PT,TAU,GT,GR,SIG,FM,ALA,ALR,ALT,ALP,ALX,CB,ANF,NP,PD,
 1    FA,KA,EL,NS)
   PRINT 5
   GO TC 2
 1 FORMAT(6X,I4)
 3 FORMAT(10A8)
30 FORMAT(10X,10A8//)
 4 FORMAT(2F6.0,2F4.0,2F6.0,7F4.0,I5,2F4.0,I1,F4.0,I2)
 5 FORMAT(1H1)
   END
```

06/20/72

```
      IF (KASE .GE. 6) 66,67
   66 PRINT 166,RNGA,ATT
      PRINT 167, SNDB
      IF (KASE .EQ. 7) 170, 171
  170 SNDB1 = PFA
      DIFF = SNDB-SNDB1
      RNGI = RNG0*10.**(DIFF*.025)
      CALL ITERAT(RNGI,ATT)
      PRINT 166, RNGI, ATT
      PRINT 167, SNDB1
  171 RETURN
   67 PRINT 150
      PRINT 151
      PRINT 152
      PRINT 153
      PRINT 14,IKASE,SNDB,ATT,RNGA
      IF (KASE .EQ. 5) 20,21
   20 DO 30 I=1,4
      CALL PDSN(PD,PFA,NPULS,I,SNDB1)
      DIFF=SNDB-SNDB1
      FAC=10.**(DIFF*.025)
      RNGI=RNG0*FAC
      CALL ITERAT(RNGI,ATT)
      PRINT 14, I,SNDB1,ATT,RNGI
   30 CONTINUE
   21 RETURN
    1 FORMAT (15X*RADAR AND TARGET PARAMETERS (INPUTS) --*/)
  100 FORMAT(15X*PULSE POWER, KW ,,,,,,,,,:,,,,,,,,,,,,,,,,,,,,*,F11,1)
  101 FORMAT(15X*PULSE LENGTH, MICROSEC ,,,,,,,,,,,,,,,,,,,,,,,*,F11,4)
  102 FORMAT(15X*TRANSMIT ANTENNA GAIN, DB :,,,,,,,,,,,,,,,,,,,*,F11,1)
  103 FORMAT(15X*RECEIVE ANTENNA GAIN, DB .:,,,,,,,,,,,,,,,,,,,*,F11,1)
  104 FORMAT(15X*FREQUENCY, MHZ ,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,*,F11,1)
  105 FORMAT(15X*RECEIVER NOISE FACTOR (FIGURE), DB ,,,,,,,,,,,*,F11,1)
  106 FORMAT(15X*BANDWIDTH CORRECTION FACTOR, DB ,,,,,,,,,,,,,,*,F11,1)
  107 FORMAT(15X*NUMBER OF PULSES INTEGRATED ,,,,,,,,,,,,,,,,,,*,6X,I5)
  108 FORMAT(15X*ANTENNA OHMIC LOSS, DB ,,,,:,,,,,,,,,,,,,,,,,,*,F11,1)
  109 FORMAT(15X*TRANSMIT TRANSMISSION LINE LOSS, DB ,,,,,,,,,,*,F11,1)
  110 FORMAT(15X*RECEIVE  TRANSMISSION LINE LOSS, DB ,,,,,,,,,,*,F11,1)
  111 FORMAT(15X*SCANNING-ANTENNA PATTERN LOSS, DB ,,,,,,,,,,,,*,F11,1)
  112 FORMAT(15X*MISCELLANEOUS LOSS, DB ,,,,,,,,,,,,,,,,,,,,,,,*,F11,1)
  113 FORMAT(15X*PROBABILITY OF DETECTION ,:,,,,,,,,,,,,,,,,,,,*,F11,3)
  114 FORMAT(15X*FALSE-ALARM PROBABILITY, NEGATIVE POWER OF TEN *,F11,1)
  115 FORMAT(15X*TARGET CROSS SECTION, SQUARE METERS ,,,,,,,,,,*,F11,4)
  116 FORMAT(15X*TARGET ELEVATION ANGLE, DEGREES ,,,,,,,,,,,,,,*,F11,2)
  117 FORMAT(15X,*SIGNAL-TO-NOISE RATIO, DB ,,,,,,,,,,,,,,,,,,,*F11,1)
   55 FORMAT(15X*MINIMUM SOLAR AND GALACTIC NOISE ASSUMED*)
   56 FORMAT(15X*AVERAGE SOLAR AND GALACTIC NOISE ASSUMED*)
   57 FORMAT(15X*MAXIMUM SOLAR AND GALACTIC NOISE ASSUMED*)
   58 FORMAT(15X*PATTERN-PROPAGATION FACTORS ASSUMED = 1.*)
   60 FORMAT(/22X,35H*********************************** /)
    2 FORMAT(15X*CALCULATED QUANTITIES (OUTPUTS) --*/)
  120 FORMAT(15X*NOISE TEMPERATURES, DEGREES KELVIN -- *)
  121 FORMAT(15X*        ANTENNA (TA) ,,,,,,,,:,,,,,,,,,,,,,,,,*,F11,1)
  122 FORMAT(15X*        RECEIVING TRANSMISSION LINE (TP) ,,,,,*,F11,1)
  123 FORMAT(15X*        RECEIVER (TE) ,,,,,,,:,,,,,,,,,,,,,,,,*,F11,1)
 1123 FORMAT(15X*        TE X LINE-LOSS FACTOR = TEI ,,,,,,,,,,*,F11,1)
```

```
                                                      06/20/72

124 FORMAT(15X*           SYSTEM (TA + TR + TEI) ,............,,......*,F11,1)
125 FORMAT(15X*TWO-WAY ATTENUATION THROUGH ENTIRE TROPOSPHERE, DB *,
   1     F7,1//)
150 FORMAT(15X* SWERLING      SIGNAL-       TROPOSPHERIC    RANGE,*)
151 FORMAT(15X* FLUCTUATION   TO-NOISE      ATTENUATION,    NAUTICAL*)
152 FORMAT(15X* CASE          RATIO, DB     DECIBELS        MILES*)
153 FORMAT(15X* ---------     -----------   ------------    --------*/)
 14 FORMAT(20X,I1,9X,F6,2, 9X,F6,2,5X,F8,1/)
166 FORMAT(/15X*RANGE = *,F8,1,* N, MI,, TROPOSPHERIC ATTENUATION = *
   1    ,F6,2,* DB*/)
167 FORMAT(15X,*FOR SPECIFIED SIGNAL-TO-NOISE RATIO = *, F6,2,* DB*//)
    END
```

06/20/72

```
      SUBROUTINE TEMP (FMHZ,ELEV,K,ANF,ALA,ALR,ATMP,TA,TR,TE,TEI,TSYS)
C     PARAMETER K DETERMINES WHAT NOISE CONDITIONS ARE ASSUMED. K=-1 IS
C     FOR QUIET SUN AND LOWEST GALACTIC NOISE (HIGH GAIN ANTENNA LOOKING
C     IN DIRECTION OF GALACTIC POLE), K=0 IS FOR AVERAGE GALACTIC NOISE
C     (GEOMETRIC MEAN OF LOWEST AND HIGHEST TEMPERATURES), AND SUN NOISE
C     TEN TIMES THE QUIET LEVEL.  K=1 IS FOR MAXIMUM GALACTIC NOISE
C     (GALACTIC CENTER, NARROW BEAM) AND SUN NOISE 100 TIMES QUIET LEVEL.
      DIMENSION T100(3)
      COMMON/RGA/ RG(75), ATTN(3,75)
      DIMENSION FR(8),TS(8)
      DATA(FR=100.,200.,300.,400.,500.,1000.,3000.,10000.),(TS=1.1E6,
     1    1.3E6,1.5E6,1.1E6,1.0E6,2.0E5,3.0E4,1.1E4)
      DATA (T100 = 500.,3050.,18650.)
      DATA (TBLKBY=2.7)
      DATA (FLAST=0.),(ELAST=100.)
      IF (FMHZ.EQ.FLAST .AND. ELEV.EQ.ELAST) GO TO 50
      FLAST=FMHZ
      ELAST=ELEV
      CALL ATLOSS(FMHZ,ELEV,ATMP)
      DO 10 I=2,8
      IF (FMHZ-FR(I))20,30,10
20    J=I-1
      TSUN = (FMHZ-FR(J))* (TS(I)-TS(J))/ (FR(I)-FR(J)) + TS(J)
      GO TO 40
30    TSUN = TS(I)
      GO TO 40
10    CONTINUE
      TSUN=1.0E4
40    ATT=10.**(-ATTN(3,75)*.05)
50    TASUN=4.75E-5*TSUN*(10.**K) * ATT
      TA1=(T100(K+2)*(100./FMHZ)**2.5+TBLKBY)*ATT+TASUN+ATMP
      ALAA=10.**(-ALA*.1)
      TA= (.876*TA1 - 254.)*ALAA + 290.
      ALRR=10.**(ALR*.1)
      TR=(ALRR-1.)*290.
      ANFF=10.**(ANF*.1)
      TE=(ANFF-1.)*290.
      TEI=TE*ALRR
      TSYS = TA + TR + TEI
      END
```

L. V. BLAKE

06/20/72

```
      SUBROUTINE ATLOSS(FMHZ,ELEV,ATMP)
      COMMON /PTR/ PP(75), TT(75), RR(75),ALPH(3,75)
      COMMON/RGA/ RG(75),ATTN(3,75)
      DIMENSION I66(4), DELH(4)
      COMMON /RRG/REFO,RAD,GRAD,U
      COMMON/DRS/ DSDH3,DRDH3,AN
      DATA(RG(1)=0.),(ATTN(1,1)=0.),(ATTN(2,1)=0.),(ATTN(3,1)=0.)
      DATA (REFO=.000313),(RAD=20898950.13),(GRAD=.00004385)
      DATA(FLAST=0.),(ELAST=100.),(CONST=.2302585)
      DATA (I66=10,14,10,3),(DELH=100.,1000.,2000.,5000.)
      ATT1(YY)=FAC2*(1.25*Y1+2.*Y2+.25*YY)
      ATT2(YY)=FAC2*(=.25*Y1+2.*Y2+1.25*YY)
      RG1(DR)=FAC1*(1.25*DRDH1+2.*DRDH2-.25*DR)
      RG2(DR)=FAC1*(-.25*DRDH1+2.*DRDH2+1.25*DR)
      IF (FMHZ.EQ.FLAST.AND.FLEV.EQ.ELAST) RETURN
      ELAST = ELEV
      ISIG=1
      THETA=ELEV/57.2957795
      SN=SIN(THETA)
      CS=COS(THETA)
      SS=SN*SN
      RP1=1.+REFO
      U=(RP1*SN)**2 - 2.*REFO - REFO*REFO
      IF (FMHZ.EQ. FLAST) GO TO 55
      CALL ALPHA(FMHZ)
      FLAST=FMHZ
   55 H=0.
      RNG=0.
      ATTEN1=ATTEN2=0.
      K=-1
      HMIN =0.
      IF (ELEV.EQ.0.) HMIN=1.E-9
      CALL DDH(HMIN)
      DRDH1=DRDH3
      DSDH1=DSDH3
      AN1=AN
      TP1=ALPH(3,1)*TT(1)
      TEHP = 0.
      Y1=ALPH(1,1)*DSDH1
      Y11=Y1
      Y12=ALPH(2,1)*DSDH1
      DO 60 J=1,4
      FAC1=DELH(J)/(3.*6076.1155)
      FAC2=2.*FAC1
      IMAX= I66(J)
      DO 61 I=1,IMAX
      K=K+2
      H=H+DELH(J)
      H1=H
      CALL DDH(H)
      DRDH2=DRDH3
      DSDH2=DSDH3
      AN2=AN
      H=H+DELH(J)
      CALL DDH(H)
      Y2=ALPH(1,K+1)*DSDH2
```

06/20/72

```
      Y21=Y2
      Y22=ALPH(2,K+1)*DSDH2
      Y3=ALPH(1,K+2)*DSDH3
      Y32=ALPH(2,K+2)*DSDH3
      AN3=AN
      IF (ELEV ,LT. 1, ,AND. H ,LT. 201.) 5, 6
    5 CC=CS*CS*(1,/RAD-REF0*GRAD/RP1)
      CC1=1,/(CC*6076,1155)
      PROD=2,*CC*H1
C     FOLLOWING IS APPROXIMATION REQUIRED NEAR THETA=0 AND H=0 FOR RANGE
C     CALCULATION,  RANGE IS CALCULATED THUS FOR H = 100 AND H= 200 WHEN
C     ELEVATION ANGLE IS LESS THAN 1 DEGREE,
      RNG=CC1*PROD/(SQRTF(PROD*SS)+SN)
      DS1=RNG
C     APPROXIMATE ATTENUATION IS RANGE (TWO-WAY) TIMES AVERAGE VALUE OF
C     GAMMA IN THE RANGE INTERVAL,
      ATTEN1=RNG*(ALPH(1,1) + ALPH(1,2))
      ATTEN2=RNG*(ALPH(2,1) + ALPH(2,2))
C     RADAR RANGE IS GEOMETRIC RANGE TIMES AVERAGE REFRACTIVE INDEX,
      RNG=RNG*(RF1 + 1, + REF0*EXP(-GRAD*H1)) * ,5
      ISIG=2
      GO TO 7
    6 DS=RG1(DRDH3)
      RNG=RNG+DS
      DS1= DS/((AN1+AN2)*,5)
      ATTEN1=ATTEN1+ATT1(Y3)
      Y1=Y12
      Y2=Y22
      ATTEN2=ATTEN2 + ATT1(Y32)
    7 RG(K+1)=RNG
      ATTN(1,K+1)= ATTEN1
      ATTN(2,K+1)=ATTEN2
      ATTN(3,K+1)=ATTEN1 + ATTEN2
      GO TO (10,11) ISIG
   11 PROD=2,*CC*H
      RNG=CC1*PROD/(SQRTF(PROD*SS)+SN)
      DS2=RNG-DS1
      ATTEN1=RNG*(ALPH(1,1) + ALPH(1,3))
      ATTEN2=RNG*(ALPH(2,1) + ALPH(2,3))
      RNG=RNG*(RP1 + 1, + REF0*EXP(-GRAD*H )) * ,5
      ISIG=1
      GO TO 12
   10 DS=RG2(DRDH3)
      RNG=RNG+DS
      DS2=DS/((AN2+AN3)*,5)
      Y1=Y11
      Y2=Y21
      ATTEN1= ATTEN1 + ATT2(Y3)
      Y1=Y12
      Y2=Y22
      ATTEN2= ATTEN2 + ATT2(Y32)
   12 RG(K+2)=RNG
      ATTN(1,K+2)=ATTEN1
      ATTN(2,K+2)=ATTEN2
      ATTN(3,K+2)=ATTEN1 + ATTEN2
      ALOSS=10,**(-ATTN(3,K+2)*,05)
```

```
      TP3=ALPH(3,K+2)*TT(K+2)*ALOSS
      DS1 = (RG(K+1)-RG(K))/((AN1+AN2)*.5)
      DS2 = (RG(K+2)-RG(K+1))/((AN2+AN3)*.5)
      IF (K.EQ.1) 70,71
C     APPROXIMATION EMPLOYED IN PLACE OF FIRST INTEGRATION STEP,
C     IFIED TO GIVE VALID RESULTS IN HIGH-ATTENUATION CASES,  ANALYTIC
C     APPROXIMATION STARTS AT STATEMENT 70,
   70 CEX=0.5*CONST*(ALPH(3,1)+ALPH(3,2))
      CEY=0.5*CONST*(ALPH(3,2)+ALPH(3,3))
      ALOS1=EXPF(-CEX*DS1)
      ALOS2=EXPF(-CEY*DS1)
      ALOS3=EXPF(-CEY*(DS1+DS2))
      DTEMP=(0.5/CONST)*((TT(1)+TT(2))*(1,-ALOS1) + (TT(2)+TT(3))*(ALOS2
     1   -ALOS3))
      GO TO 72
   71 ALOSS=10.**(-ATTN(3,K+1)*.05)
      TP2=ALPH(3,K+1)*TT(K+1)*ALOSS
      CALL INTGRT(DS1,DS2,TP1,TP2,TP3,DTEMP)
   72 TEMP = TEMP+DTEMP
      DRDH1=DRDH3
      TP1 = TP3
      Y1=Y11=Y3
      Y12=Y32
      AN1 = AN3
   61 CONTINUE
   60 CONTINUE
      ATMP=TEMP*CONST
      END
```

06/20/72

```
SUBROUTINE DDH(H)
COMMON /RRG/REFO,RAD,GRAD,U
COMMON/DRS/ DSDH3,DRDH3,AN
EX=REFO*EXP(-GRAD*H)
AN=1.+EX
V=EX*(2.+EX)
W1=H/RAD
W=W1*(2.+W1)
DSDH3=AN*(1.+W1)/SQRTF(U+V+W+V*W)
DRDH3=AN*DSDH3
END
```

34   L. V. BLAKE

```
      SUBROUTINE ALPHA(FMHZ)
      COMMON /PTF/ PP(75), TT(75), RR(75),ALPH(3,75)
      COMMON /H2O/RHEFAC
      DIMENSION FTRP(23),FTRM(23),DELH(4),IMAX(4)
      DATA(DELH=100.,1000.,2000.,5000.,),(IMAX=21,28,20,6)
      DATA(AL2=.1),(AL3=.27),(H1=26000.),(H2=32000.),(PWR=.5)
      DATA(CSTH2O= 4.693  E-3),(FRH2O=22.235),(DFLZEP0=17.99)
      DATA(CONST=3.7148),(CONST2=0.2960E),(CVP=.750064),(SMXI=.20846)
      DATA(FTRP=56.2648,58.4466,59.5910,60.4348,61.1506,61.8002,62.4112,
     162.9980,63.5685,64.1272,64.6779,65.2240,65.7626,66.2978,66.8313,
     267.3627,67.8923,68.4205,68.9478,69.4741,70.0000,70.5249,71.0497)
      DATA(FTRM=118.7505,62.4863,60.3061,59.1542,58.3239,57.6125,56.9082
     1,56.3634,55.7839,55.2214,54.6728,54.1294,53.5960,53.0695,52.5458,
     252.0259,51.5391,50.9945,50.4830,49.9730,49.4648,48.9582,48.4530)
      DATA (PP=
     11.01325E+3,1.00959E+3,1.00595E+3,1.00231E+3,9.98689E+2,9.95076E+2,
     19.91473E+2,9.87890E+2,9.84299E+2,9.60728E+2,9.77167E+2,9.73617E+2,
     19.70077E+2,9.66548E+2,9.63029E+2,9.59521E+2,9.56023E+2,9.52536E+2,
     19.49058E+2,9.45591E+2,9.42135E+2,9.08130E+2,8.75129E+2,8.43109E+2,
     18.12047E+2,7.81921E+2,7.52710E+2,7.24391E+2,6.96943E+2,6.70347E+2,
     16.44581E+2,6.19629E+2,5.95459E+2,5.72065E+2,5.49422E+2,5.27513E+2,
     15.06319E+2,4.85822E+2,4.66003E+2,4.46846E+2,4.28334E+2,4.10449E+2,
     13.93176E+2,3.76497E+2,3.60398E+2,3.44862E+2,3.29874E+2,3.15420E+2,
     13.01484E+2,2.75110E+2,2.50441E+2,2.27969E+2,2.07149E+2,1.86230E+2,
     11.71043E+2,1.55428E+2,1.41241E+2,1.28352E+2,1.16641E+2,1.06000E+2,
     19.63319E+1,8.75472E+1,7.95650E+1,7.23119E+1,6.57212E+1,5.97323E+1,
     15.42901E+1,4.93447E+1,4.48506E+1,3.53282E+1,2.78307E+1,2.19421E+1,
     11.73765E+1,1.38270E+1,1.10533E+1)
      DATA (TT=
     12.88160E+2,2.87952E+2,2.87764E+2,2.87566E+2,2.87368E+2,2.87169E+2,
     12.86971E+2,2.86773E+2,2.86575E+2,2.86377E+2,2.86179E+2,2.85981E+2,
     12.85783E+2,2.85505E+2,2.85787E+2,2.85188E+2,2.84990E+2,2.84792E+2,
     12.84594E+2,2.84396E+2,2.84199E+2,2.82217E+2,2.80237E+2,2.78256E+2,
     12.76276E+2,2.74296E+2,2.72314E+2,2.70337E+2,2.68357E+2,2.66378E+2,
     12.64399E+2,2.62420E+2,2.60442E+2,2.58463E+2,2.56485E+2,2.54507E+2,
     12.52529E+2,2.50551E+2,2.48574E+2,2.46597E+2,2.44620E+2,2.42643E+2,
     12.40664E+2,2.38609E+2,2.36713E+2,2.34737E+2,2.32761E+2,2.30785E+2,
     12.28809E+2,2.24859E+2,2.20909E+2,2.16960E+2,2.16660E+2,2.16660E+2,
     12.16660E+2,2.16660E+2,2.16660E+2,2.16660E+2,2.16660E+2,2.16660E+2,
     12.16660E+2,2.16660E+2,2.16660E+2,2.16660E+2,2.16660E+2,2.16660E+2,
     12.16660E+2,2.16660E+2,2.16660E+2,2.16660E+2,2.16660E+2,2.19069E+2,
     12.23602E+2,2.28134E+2,2.32664E+2)
      DATA(RR= 7.50000E   0, 7.43584E   0, 7.37179E   0, 7.30779E   0,
     1 7.24387E   0, 7.18004E   0, 7.11630E   0, 7.05267E   0, 6.98915E   0,
     1 6.92575E   0, 6.86247E   0, 6.79933E   0, 6.73633E   0, 6.67347E   0,
     1 6.61077E   0, 6.54822E   0, 6.48585E   0, 6.42364E   0, 6.36162E   0,
     1 6.29976E   0, 6.23814E   0, 5.63356E   0, 5.05440E   0, 4.50515E   0,
     1 3.98936E   0, 3.50702E   0, 3.04673E   0, 2.62039E   0, 2.23820E   0,
     1 1.90459E   0, 1.61971E   0, 1.38002E   0, 1.18128E   0, 1.01001E   0,
     1 8.64635E  -1, 7.36447E  -1, 6.29176E  -1, 5.34311E  -1, 4.52105E  -1,
     1 3.83598E  -1, 3.25824E  -1, 2.75746E  -1, 2.31430E  -1, 1.91737E  -1,
     1 1.56023E  -1, 1.23607E  -1, 9.46502E  -2, 7.08040E  -2, 5.23315E  -2,
     1 2.88439E  -2, 1.70220E  -2, 1.03901E  -2, 6.45199E  -3, 4.01284E  -3,
     1 2.36249E  -3, 1.46670E  -3, 1.05265E  -3, 9.02707E  -4, 8.40642E  -4,
     1 7.91078E  -4, 7.14400E  -4, 6.37648E  -4, 5.79962E  -4, 5.51107E  -4,
```

```
     1 5.44362E -4, 5.50P55E -4, 5.64024E -4, 5.87972E -4, 6.21432F -4,
     1 7.06726E -4, 8.73642E -4, 9.09155E -4, 7.32574E -4, 5.65102E -4,
     1 4.40224F -4)
       DATA(RHOFAC=1.)
       FGHZ=FMHZ*1.E-3
       FGHZ2=FGHZ*FGHZ
       FRATIO=FGHZ/FRH2O
       FSUM2=(FGHZ+FRH2O)**2
       FDIF2=(FGHZ-FRH2O)**2
       I=0
       H=-100.
       DG 100 J=1,4
       IM=IMAX(J)
       DG 100 K=1,IM
       I=I+1
       RG    =RR(I) + RHOFAC
       PPW=RG    *TT(I)/28P.75
       PR=PP(I) + PPW/CVP
       TSQ=TT(I)**2
       H=H+DELH(J)
       IF (H.LE.H1)   10,11
   10  AL1=.64
       GG TO 15
   11  IF (H .GT. H2) 12,13
   12  AL1=1.357
       GG TO 15
   13  AL1=.64 + .717*(H-H1)/54000.
   15  HAL1=CGNST2*AL1*PP(I)/TT(I)
       HSA1=HAL1*HAL1
       F0=HAL1/(FGHZ2+HSA1)
       SUM=0.
       DG 50 M=1,23
       AN=2.*M-1.
       AN1=AN+1.
       LNP2=AN*(2.*AN+3.)/AN1
       LNM2=AN1*(2.*AN-1.)/AN
       LN02=2.*(AN*AN+AN+1.)*(2.*AN+1.)/(AN*AN1)
       FNP=HAL1/((FTRP(M)-FGHZ )**2+HSA1)+HAL1/((FTRP(M)+FGHZ)**2+HSA1)
       FNM=HAL1/((FTRM(M)-FGHZ )**2+HSA1)+HAL1/((FTRM(M)+FGHZ)**2+HSA1)
       TERM=(FNP*LNP2+FNM*LNM2+F0*LN02)*EXPF(-2.06844*AN*AN1/TT(I))
       SUM=SUM+TERM
   50  CONTINUE
       ALPHO2 = CGNST*(PR    /TT(I)**3)*FGHZ2*SUM
       ALPH(1,1)=ALPHO2
       TRO=300./TT(I)
       DELF=DELZERO*(PPW*TRO+ PP(I)*CVP     *SHYI*TRO**.63)*1.E-3
       DELF2=DELF*DELF
       FFR2=FRATIO*DELF*(1./(FSUM2+DELF2)+1./(FDIF2+DELF2))
       ALPH22=CSTH2O*FGHZ*PPW*TRO**3.5*EXPF(2.144*(1.-TRO))*FFR2
       ALFRES=1.361E-2 * FGHZ2 * RG * PR * TT(I)**(-2.5)
       ALPH(2,1) = ALPH22 + ALFRES
C      ALPH(1,1) IS ABS. COEFF. FOR OXYGEN, ALPH(2,1) WATER VAPOR,
C      ALPH(3,1) IS TOTAL ABSORPTION COEFFICIENT
       ALPH(3,1)= ALPH(1,1) + ALPH(2,1)
  100  CONTINUE
       END
```

L. V. BLAKE

                                    06/20/72

```
 SUBROUTINE INTGRT (H1,H2,Y1,Y2,Y3,AREA)
 H12=H1*H1
 H22=H2*H2
 HH=H1*H2
 HPH=H1+H2
 AFAC=(Y1*H2 + Y3*H1-Y2*HPH)/(HH*HPH)
 AREA = (AFAC/3.)*(H22*H2 + H12*H1) + (Y3-Y2-AFAC*H22)*(H22-H12)/
1    (2.*H2)  + Y2*HPH
 END
```

06/20/72

```
      SUBROUTINE ITERAT(RMX,ATT)
C     GIVEN RADAR MAXIMUM RANGE IN NONABSORBING SPACE, RMX, THE
C     SUBROUTINE FINDS THE ATMOSPHERIC ATTENUATION ATT, FROM A TABLE
C     SUPPLIED BY ANOTHER SUBROUTINE VIA THE COMMON STATEMENT,
C     IT THEN CORRECTS THE RANGE RMX BY A FACTOR BASED ON ATT,  THIS
C     PROCEDURE IS ITERATED UNTIL SUCCESSIVE ATT VALUES DIFFER BY LESS
C     THAN 0.1 DECIBEL,
      COMMON/RGA/ RG(75), ATTN(3,75)
      ATTL=0.
    1 DO 10 I=2,75
      IF (RG(I) - RMX) 10,9,11
   11 J=I-1
      ATT=(ATTN(3,I)-ATTN(3,J))*(RMX-RG(J))/(RG(I)-RG(J)) + ATTN(3,J)
      GO TO 12
    9 ATT=ATTN(3,I)
      GO TO 12
   10 CONTINUE
      ATT=ATTN(3,75)
   12 DIFF = ATTL - ATT
      RMX = RMX * 10,**(DIFF*.025)
      IF (ABSF(DIFF) ,LT. 0.1) RETURN
      IF (RMX ,GE, RG(75)) RETURN
      ATTL = ATT
      GO TO 1
      END
```

05/12/72

```
      SUBROUTINE PDSN(PDT,PFA,NPULS,KASE,JOB)
      EXTERNAL PD
      DIMENSION DB0(5),SLOPE(5),PDFAC(5)
      COMMON /PDS/ PFLAST,NLAST,KSLAST
      DATA (PFLAST=0.),(NLAST=0),(KSLAST=-1)
      DATA (DB0=12.5,14.,14.,13.2,13.2),(SLOPE=6.,7.,8.,7.,8.),
     1    (PDFAC=4.8,20.,20.,13.,13.)
      DATA(DBMIN=-30.),(DBMAX=50.),(DB1=0.),(DB2=6.)
      IF (PDT.EQ.PDLAST) 1,20
    1 IF (PFA.EQ.PFLAST) 2,20
    2 IF (NPULS.EQ.NLAST) 3,20
    3 IF (KASE.EQ.KSLAST) RETURN
   20 PDLAST=PDT
      PFLAST=PFA
      NLAST=NPULS
      KSLAST=KASE
      K=KASE + 1
      PULS=NPULS
      DB1=DB0(K)-SLOPE(K)*ALOG10(PULS)+(PDT-.5)*PDFAC(K)+(PFA-8.)*.4-1.
      DB2=DB1+2.
      CALL INVERS(DBMIN,DBMAX,DB1,DB2,4,15,NOI,SDB,PD1,PDT,PD)
      END
```

05/12/72

```
FUNCTION PD(SNDB)
COMMON/POS/FA,N,KASE
NP=N
FAN=FA
KAS=KASE
CALL MARSWR (SNDB,NP,FAN,KAS,PD1)
PD=PD1
END
```

05/12/72

```
      SUBROUTINE INVERS(XMIN,XMAX,XLO,XHI,NSIG,LIM,NOI,X,F1,FT,F)
C     THIS IS A MODIFICATION (APRIL 1970) OF A PREVIOUS SUBROUTINE NAMED
C     INVERT. THIS VERSION HAS ADDITIONAL PARAMETERS. L. V. BLAKE, NRL.
C   THIS SUBROUTINE FINDS VALUE OF X THAT RESULTS IN F(X) = FT, BY
C   ITERATION BASED ON LINEAR INTERPOLATION/EXTRAPOLATION FROM PREVIOUS
C   TWO TRIALS.  FUNCTION F MUST BE MONOTONIC.
      TEST = 10.**(-NSIG)
      FD = FT
      IF (FT .EQ. 0.) FD = 1.
      NOI = 1
      DELTA = XHI - XLO
      X1=XLO
      X2=XHI
      F1=F(X1)
      F2=F(X2)
      SLOPE=(F2-F1)/(X2-X1)
      IF (SLOPE .EQ. 0.) 10,21
   10 FMAX=F(XMAX)
      FMIN=F(XMIN)
      SLOPE = (FMAX-FMIN)/(XMAX-XMIN)
   21 IF((F2-FT)*SLOPE .LT. 0.) 22,23
   22 X1=X2
      F1=F2
      X2=X2+DELTA
      IF (X2 .GT. XMAX) X2=XMAX
      F2=F(X2)
      GO TO 21
   23 IF((FT-F1)*SLOPE .LT. 0.) 24,25
   24 X2=X1
      X1=X1-DELTA
      IF (X1 .LT. XMIN) X1=XMIN
      F2=F1
      F1=F(X1)
      GO TO 23
   25 XA=X1
      XB=X2
      FA=F1
      FB=F2
      IF (ABSF(F2-FT) .LT. ABSF(F1-FT)) 7,6
    7 F22=F2
      F2=F1
      F1=F22
      X22=X2
      X2=X1
      X1=X22
      GO TO 6
    1 F1=F(X)
      X1=X
      TEST1 = ABS((F1-FT)/FD)
      IF (TEST1 .LE. TEST) 2,6
    2 RETURN
    6 IF (NOI .GE. LIM) 12,13
   12 PRINT 40
      PRINT 41, LIM
      PRINT 42, XMIN, XMAX, XLO, XHI, NSIG, LIM, NOI, X, F1, FT
      RETURN
```

11

05/12/72

```
13 IF (F1 .EC. F2) 15,16
15 IF (F1 .FC. FB) 17,18
17 XB=X1
   X2=XB
   GO TC 19
18 XA=X1
   X2=XA
19 X=(XA+XB)*.5
   GO TO 1
16 X=(X1-X2)*(FT-F2)/(F1-F2) + X2
   IF (X .LT. XA) X=XA
   IF (X .GT. XB) X=XB
   NOI = NOI + 1
   F2=F1
   X2=X1
   GO TO 1
40 FORMAT (//*   MESSAGE FROM SUBROUTINE INVERS -- *//)
41 FORMAT (*   FUNCTION INVERSION NOT ACCOMPLISHED WITHIN SPECIFIED *
  1  ,I3, * ITERATIONS.*/)
42 FORMAT (/*   INVERS PARAMETERS WERE *,4(E10.3,2X),3(I3,2X),E10.3,
  1  2(2X,E10.3) //)
   END
```

05/12/72

```
      SUBROUTINE MARSWR (SNDB,N,FA,KASE,PN)

C
C     INPUTS ARE -- SNDB, SIGNAL-TO-NOISE POWER RATIO IN DECIBELS --
C     N, NUMBER OF PULSES INTEGRATED --
C     FA, FALSE ALARM PROBABILITY, EXPRESSED AS ABSOLUTE VALUE OF POWER
C     OF TEN (E.G., FA = 8. MEANS 10.**(-8.) FALSE ALARM PROBABILITY --
C     KASE, SWERLING FLUCTUATION MODEL, WITH KASE = 0 FOR NONFLUCTUATING
C     OUTPUT PN IS PROBABILITY OF DETECTION
C
C     BASED ON PROGRAM WRITTEN ATJHU APPLIED PHYSICS LABORATORY, NAMED
C     SUBROUTINE MARCUM.  MODIFIED AT NRL BY L. V. BLAKE.  THIS VERSION
C     DATED APRIL 1971
C     APL VERSION DEFINED FA AS FALSE ALARM NUMBER (MARCUM CONCEPT).
C     NRL MOD CHANGED THIS TO FALSE ALARM PROBABILITY (AS DEFINED ABOVE)
C     SOME OTHER CHANGES ALSO.
C
      DOUBLE PRECISION ENPR,YBPR,GAMPR,PYB,H,Y0,E0,Y1,E1,STEP,YB    00000200
      DOUBLE PRECISION DGAM, DEVAL, SUMLOG, SUML, FAN, EN           00000300
C                                                                  00000400
C     COMPUTE MARCUM-SWERLING DETECTION PROBABILITIES              00000500
C                                                                  00000600
C     CONVERT S/N IN DB TO NUMERICAL S/N RATIO                     00000700
C                                                                  00000800
      SNR = 10.**(SNDB*.1)                                         00000900
C                                                                  00001000
C     TO CONVERT THIS SUBROUTINE TO INTERPRET FA AS THE MARCUM FALSE-
C     ALARM NUMBER, CHANGE THE NEXT STATEMENT TO READ -- MODE = 0
C
      MODE=1
C     IF MODE IS 1, CONVERT FA  TO MEAN EXPONENT OF FALSE-ALARM    00001100
C     PROBABILITY RATHER THAN MARCUM FALSE-ALARM NUMBER            00001200
C                                                                  00001300
      IF (MODE) 800, 800, 900                                      00001400
  900 FAN =DLOG10(DLOG(.5)/DLOG(1.-(10.0)**(-FA)))                 00001500
      GO TO 905                                                    00001600
  800 FAN = FA                                                     00001700
C                                                                  00001800
C     TEST INPUTS                                                  00001900
C                                                                  00002000
  905 IF(N) 99,99,2                                                00002100
    2 IF(FA)99,99,3                                                00002200
    3 IF(KASE) 99,4,4                                              00002300
    4 IF(KASE-4) 5,5,99                                            00002400
C                                                                  00002500
C     ESTIMATE BIAS LEVEL                                          00002600
C                                                                  00002700
    5 ENPR = 0.                                                    00002800
    6 ENPR = FAN                                                   00002900
      EN = N                                                       00003000
      YBPR = 0.                                                    00003100
      IF (NPREV .EQ. N .AND. FAPREV .EQ. FA) GO TO 777
      IF(N-12) 7,7,8                                               00003200
    7 YBPR = EN*(1.+2.2*ENPR/EN**((2.0/3.0)+.015*ENPR))            00003300
      GO TO 11                                                     00003400
    8 YBPR = EN*(1.+1.3*ENPR/EN**(.5+.011*ENPR))                   00003500
C                                                                  00003600
```

05/12/72

```
C       COMPUTE BIAS LEVEL                              00003700
C                                                       00003800
   11 ENPR = 10.**ENPR                                  00003900
      GAMPR = DGAM(YBPR,N-1)                            00004000
      PYB = .5**(1./ENPR)                               00004100
      SUML = SUMLOG(N-1)                                00004200
      IF(GAMPR-PYB) 10,12,12                            00004300
   10 H = .01                                           00004400
      GO TO 14                                          00004500
   12 H = -.01                                          00004600
   14 YO = YBPR                                         00004700
      FO = DEVAL(YO,N-1,SUML)                           00004800
   16 Y1 = YO+H                                         00004900
      E1 = DEVAL(Y1,N-1,SUML)                           00005000
      STEP = GAMPR + H*(E0+E1)/2.                       00005100
      IF(SIGNF(1.,STEP-PYB)-SIGNF(1.,H)) 18,20,18       00005200
   18 YO = Y1                                           00005300
      FO = F1                                           00005400
      GAMPR = STEP                                      00005500
      GO TO 16                                          00005600
   20 IF(H) 22,24,24                                    00005700
   22 YB = Y1 - H*(PYB-STEP)/(GAMPR-STEP)               00005800
      GO TO 30                                          00005900
   24 YB = YO + H*(PYB-GAMPR)/(STEP-GAMPR)              00006000
   30 BIAS = YB                                         00006100
  777 YB = BIAS
      NPREV = N
      FAPREV = FA
C                                                       00006200
C                                                       00006300
C       SELECT M-S CASE                                 00006400
      X = SNR                                           00006500
      K = KASE+1                                        00006600
      GO TO (100,200,300,400,500), K                   00006700
C                                                       00006800
C       CASE 0                                          00006900
C                                                       00007000
  100 SUM = 0.                                          00007100
      P = EN*X                                          000072C0
      IF(YB-P-EN) 150,102,102                           00007300
  102 KS = -(EN+1.)/2. + SQRTF(((EN-1.)/2.)**2+P*YB)    00007400
      KS = XMAXOF(KS,0)                                 00007500
      GS = 1.-GAM(YB,KS+N-1,TN)                         00007600
      TS = EVAL(P,KS)*GS                                00307700
      U = GS                                            00007800
      X = KS                                            00007900
      TERM = TS                                         00008000
      TL = TN                                           00008100
  110 TEMP = SUM+TERM                                   00008200
      IF(SUM-TEMP) 112,116,116                          00008300
  112 SUM = TEMP                                        0000BAC0
      IF(K) 116,116,114                                 00008500
  114 TERM = TERM*FLOATF(K)*(G-TL)/(P*G)                00008600
      G = G-TL                                          00008700
      K = K-1                                           00008800
      TL = TL*FLOATF(K+N)/YB                            00008900
```

44   L. V. BLAKE

05/12/72

```
      GO TC 110                                              00009000
  116 TL = TN*YB/FLOATF(KS*N)                                00009100
      K = KS+1                                               00009200
      G = GS*TL                                              00009300
      TERM = TS*P*G/(GS*FLOATF(K))                           00009400
  120 TEMP = SUM+TERM                                        00009500
      IF(SUM-TEMP) 122,190,190                               00009600
  122 SUM = TEMP                                             00009700
      TL = TL*YB/FLOATF(K+N)                                 00009800
      K = K+1                                                00009900
      TERM = TERM*P*(G+TL)/(G*FLOATF(K))                     00010000
      G = G+TL                                               00010100
      GO TC 120                                              00010200
  150 KS = -1. - EN/2. + SQRTF(EN**2/4.+P*YB)                00010300
      KS =XMAXOF(KS,0)                                       00010400
      GS = GAM(YB,KS+N-1,TN)                                 00010500
      IF(GS) 174,174,155                                     00010600
  155 TS = EVAL(P,KS)*GS                                     00010700
      G = GS                                                 00010800
      TERM = TS                                              00010900
      K = KS                                                 00011000
      TL = TN                                                00011100
  160 TEMP = SUM+TERM                                        00011200
      IF(SUM-TEMP) 162,166,166                               00011300
  162 SUM = TEMP                                             00011400
      IF(K) 166,166,164                                      00011500
  164 TERM = TERM*FLOATF(K)*(G+TL)/(P*G)                     00011600
      G = G+TL                                               00011700
      TL = TL*FLOATF(K+N-1)/YB                               00011800
      K = K-1                                                00011900
      GO TC 160                                              00012000
  166 TL = TN*YB/FLOATF(KS+N)                                00012100
      K = KS+1                                               00012200
      G = GS-TL                                              00012300
      TERM = TS*P*G/(GS*FLOATF(K))                           00012400
  170 TEMP = SUM + TERM                                      00012500
      IF(SUM-TEMP) 172,174,174                               00012600
  172 SUM = TEMP                                             00012700
      TL = TL*YB/FLOATF(K+N)                                 00012800
      TERM = TERM*P*(G-TL)/(G*FLOATF(K+1))                   00012900
      G = G-TL                                               00013000
      K = K+1                                                00013100
      GO TC 170                                              00013200
  174 SUM = 1.-SUM                                           00013300
  190 PN = SUM                                               00013400
      GO TC 90                                               00013500
C                                                            00013600
C     CASE 1                                                 00013700
C                                                            00013800
  200 IF(N-1) 210,210,220                                    00013900
  210 PN = EXPF(-YB/(1.+X))                                  00014000
      GO TC 90                                               00014100
  220 TEMP = 1. + 1./(EN*X)                                  00014200
      PN = 1. - GAM(YB,N-2,DUM) + EXPF((EN-1.)*LOGF(TEMP)-YB/(1.+EN*X))  00014300
     1                      *GAM(YB/TEMP,N-2,DUM)            00014400
      GO TC 90                                               00014500
```

05/12/72

```
C
C       CASE 2                                        00014600
C                                                     00014700
  300 IF(N-1) 310,310,320                             00014800
  310 PN = EXPF(-YB/(1.+X))                           00014900
      GO TO 90                                        00015000
  320 PN = 1. - GAM(YB/(1.+X),N-1,DUM)                00015100
      GO TO 90                                        00015200
C                                                     00015300
C       CASE 3                                        00015400
C                                                     00015500
  400 IF(N-2) 410,420,430                             00015600
  410 PN = (1.+2.*X*YB/(X+2.)**2)*EXPF(-2.*YB/(2.+X)) 00015700
      GO TO 90                                        00015800
  420 PN = (1.+YB/(1.+X))*EXPF(-YB/(1.+X))            00015900
      GO TO 90                                        00016000
  430 C = 2./(2.+EN*X)                                00016100
      D = 1.-C                                        00016200
      IF(YB*D-EN) 440,450,450                         00016300
  440 SUM = 0.                                        00016400
      TERM = 1.                                       00016500
      J = N                                           00016600
  442 TEMP = SUM+TERM                                 00016700
      IF(SUM-TEMP) 444,446,446                        00016800
  444 SUM = TEMP                                      00016900
      TERM = TERM*YB*D/FLOATF(J)                      00017000
      J = J+1                                         00017100
      GO TO 442                                       00017200
  446 PN = 1. - GAM(YB,N-2,DUM) + C*YB*EVAL(YB,N-2)   00017300
     1          + D*EVAL(YB,N-1)*(1.+C*YB-(EN-2.)*C/D)*SUM 00017400
      GO TO 90                                        00017500
  450 PN = 1. - GAM(YB,N-3,DUM) + YB*EVAL(YB,N-3)*C/D 00017600
     1          + EXPF(-C*YB-(EN-2.)*LOGF(D))*(1.+C*YB-(EN-2.)*C/D) 00017700
     2          *GAM(YB*D,N-3,DUM)                    00017800
      GO TO 90                                        00017900
C                                                     00018000
C       CASE 4                                        00018100
C                                                     00018200
  500 SUM = 0.                                        00018300
      C = 2./(2.+X)                                   00018400
      D = 1.-C                                        00018500
      Q = C/D                                         00018600
      P = C*YB                                        00018700
      KS = (3.*EN+(YB*D))/2.-SQRTF((EN-1.+(YB*D))**2/4.+(YB*D)*(EN+1.)) 00018800
      KS = XMINOF(KS,N)                               00018900
      KS = XMAXOF(KS,0)                               00019000
      K = KS                                          00019100
      J = N-KS                                        00019200
      FKS = KS                                        00019300
      K = XMINOF(KS,N)                                00019400
      IF(YB-EN*(1.+D)) 550,501,501                    00019500
  501 GS = 1. - GAM(P,2*N-1-KS,TN)                    00019600
      IF(GS) 526,526,502                              00019700
  502 TS = EXPF(FKS*LOGF(C)+(EN-FKS)*LOGF(D)+SUMLOG(N)-SUMLOG(KS) 00019800
     1          -SUMLOG(J)+LOGF(GS))                  00019900
      G = GS                                          00020000
                                                      00020100
```

05/12/72

```
        TERM = TS                                                        00020200
        TL = TN                                                          00020300
510 TEMP = SUM+TERM                                                      00020400
        IF(SUM-TEMP) 512,516,516                                         00020500
512 SUM = TEMP                                                           00020600
        IF(K) 516,516,514                                                00020700
514 TL = TL*P/FLOATF(2*N-K)                                              00020800
        TERM = TERM*FLOATF(K)*(G+TL)/(Q*FLOATF(N-K+1)*G)                 00020900
        G = G+TL                                                         00021000
        K = K-1                                                          00021100
        GO TC 510                                                        00021200
516 IF(KS-N) 518,526,526                                                 00021300
518 TERM = TS*Q*FLOATF(N-KS)*(GS-TN)/(FLOATF(KS+1)*GS)                   00021400
        G = GS-TN                                                        00021500
        TL = TN*FLOATF(2*N-1-KS)/P                                       00021600
        K = KS+1                                                         00021700
520 TEMP = SUM+TERM                                                      00021800
        IF(SUM-TEMP) 522,526,526                                         00021900
522 SUM = TEMP                                                           00022000
        IF(K-N) 524,526,526                                              00022100
524 TERM = TERM*Q*FLOATF(N-K)*(G+TL)/(FLOATF(K+1)*G)                     00022200
        G = G+TL                                                         00022300
        TL = TL*FLOATF(2*N-1-K)/P                                        00022400
        K = K+1                                                          00022500
        GO TC 520                                                        00022600
526 PN = SUM                                                             00022700
        GO TC 90                                                         00022800
550 GS = GAM(P,2*N-1-KS,TN)                                              00022900
        IF(GS) 576,576,552                                               00023000
552 TS = EXPF(FKS*LOGF(C)+(EN-FKS)*LOGF(D)+SUMLOG(N)-SUMLOG(KS)         00023100
     1        -SUMLOG(J)+LOGF(GS))                                       00023200
        G = GS                                                           00023300
        TERM = TS                                                        00023400
        TL = TN                                                          00023500
560 TEMP = SUM+TERM                                                      00023600
        IF(SUM-TEMP) 562,566,566                                         00023700
562 SUM = TEMP                                                           00023800
        IF(K) 566,566,564                                                00023900
564 TL = TL*P/FLOATF(2*N-K)                                             00024000
        TERM = TERM*FLOATF(K)*(G-TL)/(Q*FLOATF(N-K+1)*G)                00024100
        G = G-TL                                                         00024200
        K = K-1                                                          00024300
        GO TC 560                                                        00024400
566 IF(KS-N) 568,576,576                                                 00024500
568 TERM = TS*Q*FLOATF(N-KS)*(GS+TN)/(FLOATF(KS+1)*GS)                   00024600
        G = GS+TN                                                        00024700
        TL = TN*FLOATF(2*N-1-KS)/P                                       00024800
        K = KS+1                                                         00024900
570 TEMP = SUM+TERM                                                      00025000
        IF(SUM-TEMP) 572,576,576                                         00025100
572 SUM = TEMP                                                           00025200
        IF(K-N) 574,576,576                                              00025300
574 TERM = TERM*Q*FLOATF(N-K)*(G+TL)/(FLOATF(K+1)*G)                     00025400
        G = G+TL                                                         00025500
        TL = TL*FLOATF(2*N-1-K)/P                                        00025600
        K = K+1                                                          00025700
```

05/12/72

```
        GO TC 570                                              00025800
    576 PN = 1.-SUM                                            00025900
        GO TO 90                                               00026000
                                                               00026100
   C                                                           00026200
   C    SET PROBABILITY                                        00026300
   C                                                           00026400
     90 IF(PN) 91,94,92                                        00026500
     91 PN = 0.                                                00026600
        GO TC 94                                               00026700
     92 IF(PN-1.) 94,94,93                                     00026800
     93 PN = 1.                                                00026900
     94 RETURN                                                 00027000
   C                                                           00027100
   C    ERROR MESSAGE FOR BAD INPUTS                           00027200
   C                                                           00027300
     99 WRITE (61,9)   N,FA ,SNR,KASE                          00027400
      9 FORMAT (1H0 /50H UNREASONABLE CALL SEQUENCE TO MARCUM, ZERO RESULT00027400
        1              7HS GIVEN //4H N = I8,5X,5HFA  = F16.8,5X,5HSNR =  00027500
        2              E16.8,5X,6HKASE = I8)                   00027600
                                                               00027700
        PN = 0.                                                00027800
        BIAS = 0.                                              00027900
        RETURN                                                 00028000
        END
```

48　　　　　　　　　　　　　　L. V. BLAKE

05/12/72

```
                                                          0002B100
      FUNCTION DGAM(B,N)                                  0002B200
      DOUBLE PRECISION  SUM,  TERM,TEMP,FJ,DGAM, DEVAL, B, SUML,SUMLOG
C             INTEGRAL = 1-(SUM, J=0 TO N, OF EXPF(J*LOGF(B)-B-LOGF(NFAC((( 28300
      SUM = 0.                                            (((28400
      K = 8                                               (((28500
      IF(K-N) 100,200,200                                 (((28600
  100 J = N+1                                             (((28700
      SUML = SUMLOG(J)                                    (((28800
      TERM = DEVAL(B,J,SUML)                              (((28900
   10 TEMP = SUM+TERM                                     (((29000
      IF(SUM-TEMP) 15,20,20                               (((29100
   15 SUM = TEMP                                          (((29200
      J = J+1                                             (((29300
      FJ = J                                              (((29400
      TERM = TERM*B/FJ                                    (((29500
      GO TO 10                                            (((29600
   20 DGAM = SUM                                          (((29700
      RETURN                                              (((29800
  200 J = N                                               (((29900
      SUML = SUMLOG(J)                                    (((30000
      TERM = DEVAL(B,J,SUML)                              (((30100
   30 TEMP = SUM+TERM                                     (((30200
      IF(SUM-TEMP) 35,40,40                               (((30300
   35 SUM = TEMP                                          (((30400
      IF(J-1) 40,36,36                                    (((30500
   36 FJ = J                                              (((30600
      TERM = TERM*FJ/B                                    (((30700
      J = J-1                                             (((30800
      GO TO 30                                            (((30900
   40 DGAM = 1.-SUM                                       (((31000
      RETURN                                              (((31100
      END                                                 (((31200
```

05/12/72

```
      FUNCTION DEVAL (Y,N,SUML)                              (((31300
      DOUBLE PRECISION  XPON,EN,DEVAL,  Y,SUML               (((31400
      XPON = -Y                                              (((31500
      IF(N) 20,20,10                                         (((31600
   10 EN = N                                                 (((31700
      XPON = XPON+EN*DLOG(Y)-SUML                            (((31800
   20 DEVAL = DEXP(XPON)                                     (((31900
      RETURN                                                 (((32000
      END                                                    (((32100
```

05/12/72

```
      FUNCTION GAM(B,N,TN)                              (((32200
C                SINGLE PRECISION VERSION OF CGAM       (((32300
      SUM = 0.                                          (((32400
      K = B                                             (((32500
      IF(K-N) 100,200,200                               (((32600
100   J = N+1                                           (((32700
      TERM = EVAL(B,J)                                  (((32800
      TN = TERM*FLOATF(J)/B                             (((32900
10    TEMP = SUM+TERM                                   (((33000
      IF(SUM-TEMP) 15,20,20                             (((33100
15    SUM = TEMP                                        (((33200
      J = J+1                                           (((33300
      FJ = J                                            (((33400
      TERM = TERM*B/FJ                                  (((33500
      GO TO 10                                          (((33600
20    GAM = SUM                                         (((33700
      RETURN                                            (((33800
200   J = N                                             (((33900
      TERM = EVAL(B,J)                                  (((34000
      TN = TERM                                         (((34100
30    TEMP = SUM+TERM                                   (((34200
      IF(SUM-TEMP) 35,40,40                             (((34300
35    SUM = TEMP                                        (((34400
      IF(J-1) 40,36,36                                  (((34500
36    FJ = J                                            (($34600
      TERM = TERM*FJ/B                                  ((¿34700
      J = J-1                                           (((34800
      GO TO 30                                          (((34900
40    GAM = 1.-SUM                                      (((35000
      RETURN                                            (((35100
      END                                               (((35200
```

05/12/72

```
     FUNCTION EVAL(Y,N)                              (((35300
     XPON = -Y                                       (((35400
     IF (N) 20,20,10                                 (((35500
 10  EN = N                                          (((35600
     XPON = XPON+EN*LOGF(Y)-SUMLOG(N)                (((35700
 20  EVAL = EXPF(XPON)                               (((35800
     RETURN                                          (((35900
     END                                             (((36000
```

52   L. V. BLAKE

05/12/72

```
      FUNCTION SUMLOG(N)                                          (((36100
      DOUBLE PRECISION A, B, SUMLOG                               (((36200
      DIMENSION A(1000)
      DATA(DUMA = 0.),(DUMB = 0.)                                 (((36400
      NMAX=1000
      IF(DUMA-DUMB) 20,10,20                                      (((36600
   10 DUMA = 1.                                                   (((36700
      DUMB = 0.                                                   (((36800
      NLAST = 1                                                   (((36900
      A(1) = 0.                                                   (((37000
   20 NN = XABSF(N)                                               (((37100
      IF(NN-1) 30,30,40                                           (((37200
   30 SUMLOG = 0.                                                 (((37300
      RETURN                                                      (((37400
   40 IF(NN-NLAST) 50,50,60                                       (((37500
   50 SUMLOG = A(NN)                                              (((37600
      RETURN                                                      (((37700
   60 K = NLAST+1                                                 (((37800
      IF(NN-NMAX) 70,70,80                                        (((37900
   70 DO 72 I=K,NN                                                (((38000
   72 A(I) = A(I-1) + DLOG(FLOATF(I))                             (((38100
      NLAST = NN                                                  (((38200
      GO TO 50                                                    (((38300
   80 IF(NLAST-NMAX) 82,90,90                                     (((38400
   82 DO 84 I=K,NMAX                                              (((38500
   84 A(I) = A(I-1) + DLOG(FLOATF(I))                             (((38600
      NLAST = NMAX                                                (((38700
   90 B = A(NMAX)                                                 (((38800
      K = NMAX+1                                                  (((38900
      DO 92 I=K,NN                                                (((39000
   92 B = B + DLOG(FLOATF(I))                                     (((39100
      SUMLOG = B                                                  (((39200
      RETURN                                                      (((39300
      END                                                         (((39400
```

UNCLASSIFIED

REFERENCES

1. Blake, L.V., "A Guide to Basic Pulse-Radar Maximum-Range Calculation," Part 1, NRL Report 6930, and Part 2, NRL Report 7010, Dec. 1969.

2. Bean, B.R., and Dutton, E.J., "Radio Meteorology," National Bureau of Standards Monograph 92, U.S. Government Printing Office, Washington, D.C., 1966.

3. Blake L.V., "Machine Plotting of Radio/Radar Vertical-Plane Coverage Diagrams," NRL Report 7098, June 1970.

4. Boothe, R.R., "A Digital Computer Program for Determining the Performance of an Acquisition Radar through Application of Radar Detection Probability Theory," U.S. Army Missile Command Report No. RD-TR-64-2, Dec. 1964.

5. White, D.M., "Radar Simulation and Analysis by Digital Computer," Johns Hopkins University Applied Physics Laboratory Report TG-952, Jan. 1968.

6. Fehlner, L.F., "Marcum and Swerling's Data on Target Detection by a Pulsed Radar," Johns Hopkins University Applied Physics Laboratory Report TG-451, July 1962.

7. Kirkwood, P.K., "Radar Cumulative Detection Probabilities for Radial and Nonradial Target Approaches," Rand Corporation Report RM-4643-PR, Sept. 1965.

8. Nolen, J.C., and others, "Statistics of Radar Detection," Bendix Corporation, Baltimore, Md., Feb. 1966.

9. Killinger, D.K., "Computer Aided Radar Design (CARD)," Naval Avionics Facility Interim Report TR 1461, 12 Aug. 1969.

10. Minzner, R.A., Ripley, W.S., and Condron, T.P., "U.S. Extension to the ICAO Standard Atmosphere," U.S. Dept. of Commerce Weather Bureau and USAF ARDC Cambridge Research Center, Geophysics Research Directorate, U.S. Government Printing Office, Washington, D.C., 1958.

11. Sissenwine, N., Grantham, D.D., and Salmela, H.A., "Humidity up to the Mesopause," Air Force Cambridge Research Laboratories Report AFCRL-68-0550 (Air Force Surveys in Geophysics, No. 206), Oct. 1968.

12. Blake, L.V., "A Modified Simpson's Rule and Fortran Subroutine for Cumulative Numerical Integration of a Function Defined by Data Points," NRL Memorandum Report 2231, Apr. 1971.

13. Meeks, M.L., and Lilley, A.E., "The Microwave Spectrum of Oxygen in the Earth's Atmosphere," J. Geophys. Res. 68 (No. 6):1683-1703 (Mar. 15, 1963).

14. Reber, E.E., Mitchell, R.L., and Carter, C.J., "Attenuation of the 5-mm Wavelength Band in a Variable Atmosphere," IEEE Transactions AP-18 (No. 4):472-479 (July 1970).

15. Blake, L.V., "Radio Ray (Radar) Range-Height-Angle Charts," NRL Report 6650, Jan. 22, 1968, and Microwave J. 11 (No. 10):49-53 (Oct. 1968).

16. Blake, L.V., "Ray Height Computation for a Continuous Nonlinear Atmospheric Refractive-Index Profile," Radio Science 3 (new series) (No. 1):85-92 (Jan. 1968).

Appendix A

A FORM FOR LISTING DATA CARD INPUTS TO PROGRAM RGCALC

The following form has been devised for listing the input quantities needed on data cards for Program RGCALC. The form provides a convenient reference for punching the data cards, since it indicates the card columns for each input quantity. The "case" options (1 through 7) are also described in a footnote on the form.

DATA CARD FORMATS FOR PROGRAM RGCALC
(Reference: NRL Report 7448)

Cards 1, 3, 5, 7 ...: Name of radar (Format 10A8). Cols. 1-80. (Any other descriptive alphanumeric material can be entered in these columns.)

Cards 2, 4, 6, 8 ...: Radar data as below. (Note: A quantity with F format can be punched without decimal point if it has an integer value and is right-adjusted. Non-integer numbers must be punched with decimal point; right-adjustment then unnecessary. Integer-format quantities must be right-adjusted in specified column field.)

DATA ITEM	FORMAT SPEC.	COLS.	RADAR NAME	RADAR NAME	RADAR NAME
Transmitter power, kW (P_t)	F6.0	1-6			
Pulse length, μsec (τ)	F6.0	7-12			
Transmit antenna gain, dB (G_t)	F4.0	13-16			
Receive antenna gain, dB (G_r)	F4.0	17-20			
Target cross section, sq m (σ)	F6.0	21-26			
Frequency, MHz (f)	F6.0	27-32			
Antenna ohmic loss, dB (L_a)	F4.0	33-36			
Receiving line loss, dB (L_r)	F4.0	37-40			
Transmit line loss, dB (L_t)	F4.0	41-44			
Antenna-pattern scan loss, dB (L_p)	F4.0	45-48			
Miscellaneous loss, dB (L_x)	F4.0	49-52			
Bandwidth/shape factor, dB (C_B)	F4.0	53-56			
Receiver noise factor, dB (NF)	F4.0	57-60			
Number of pulses (M) See Report 6930, Eqs. (71) and (72)	I5	61-65			
Cases 0-5: Probability of detection Cases 6-7: Signal-to-noise ratio, dB	F4.0	66-69			
Cases 0-5: False alarm exp't ($-\log_{10} P_{fa}$) Case 6: Blank; Case 7, Signal/noise, dB	F4.0	70-73			
Case (0 to 7)*	I1	74			
Target elevation angle, degrees	F4.0	75-78			
Galactic-noise-level code (−1, 0, +1) (minimum, average, maximum)	I2	79-80			

*Cases are: 0 to 4 – range is calculated for corresponding Swerling fluctuation case; 5 – calculated for all five Swerling cases (0, 1, 2, 3, 4); 6 – calculated for the S/N value in Cols. 66-69, 7 – calculated for two S/N values, one in 66-69, other in 70-73.